

REPORT DOCUMENTATION PAGE			Form Approved OMB NO. 0704-0188		
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA, 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) 27-08-2012		2. REPORT TYPE Final Report		3. DATES COVERED (From - To) 24-Aug-2007 - 23-Apr-2013	
4. TITLE AND SUBTITLE Investigation of Terrain Analysis and Classification Methods for Ground Vehicles			5a. CONTRACT NUMBER W911NF-07-1-0540		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER 611102		
6. AUTHORS Karl Iagnemma			5d. PROJECT NUMBER		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAMES AND ADDRESSES Massachusetts Institute of Technology (MIT) Office of Sponsored Programs Bldg. E19-750 Cambridge, MA 02139 -4307			8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Office P.O. Box 12211 Research Triangle Park, NC 27709-2211			10. SPONSOR/MONITOR'S ACRONYM(S) ARO		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S) 52941-EV.7		
12. DISTRIBUTION AVAILABILITY STATEMENT Approved for Public Release; Distribution Unlimited					
13. SUPPLEMENTARY NOTES The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.					
14. ABSTRACT Unmanned ground vehicles (UGVs) must rapidly and robustly characterize the nature of the terrain they are traversing, to improve autonomous mobility. This research program has focused on the development of a framework for self-supervised terrain classification, which allows a UGV to automatically learn the properties of terrain without human guidance. Work has also focused on novel applications of the self-supervised terrain learning approach, including urban/semi-urban driving on road networks. Finally, research has led to the development of					
15. SUBJECT TERMS Ground vehicles, terramechanics, terrain sensing					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Karl Iagnemma
a. REPORT UU	b. ABSTRACT UU	c. THIS PAGE UU			19b. TELEPHONE NUMBER 617-452-3262

Report Title

Investigation of Terrain Analysis and Classification Methods for Ground Vehicles

ABSTRACT

Unmanned ground vehicles (UGVs) must rapidly and robustly characterize the nature of the terrain they are traversing, to improve autonomous mobility. This research program has focused on the development of a framework for self-supervised terrain classification, which allows a UGV to automatically learn the properties of terrain without human guidance. Work has also focused on novel applications of the self-supervised terrain learning approach, including urban/semi-urban driving on road networks. Finally, research has led to the development of novel sensing techniques for analyzing robot-terrain interaction mechanics at the micro scale.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

<u>Received</u>	<u>Paper</u>
-----------------	--------------

TOTAL:

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

<u>Received</u>	<u>Paper</u>
-----------------	--------------

TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations

Number of Presentations: 0.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

<u>Received</u>	<u>Paper</u>
-----------------	--------------

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Peer-Reviewed Conference Proceeding publications (other than abstracts):

<u>Received</u>	<u>Paper</u>
-----------------	--------------

08/27/2012	6.00	Carmine Senatore, Markus Wulfmeier, Paramsothy Jayakumar, Karl Iagnemma. INVESTIGATION OF STRESS AND FAILURE IN GRANULAR SOILS FOR LIGHTWEIGHT ROBOTIC VEHICLE APPLICATIONS, 2012 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM. 2012/08/14 00:00:00, . : ,
------------	------	--

TOTAL:	1
---------------	----------

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

Received Paper

03/11/2011	2.00	S. Zhou, K. Iagnemma. Self-supervised Learning Method for Unstructured Road Detection using Fuzzy Support Vector Machines, (03 2011)
03/11/2011	1.00	M. McDaniel, T. Nishihata, C. Brooks, K. Iagnemma. Ground Plane Identification Using LIDAR in Forested Environments, (03 2011)
03/11/2011	3.00	S. Zhou, J. Xi, M. McDaniel, T. Nishihata, P. Salesses, K. Iagnemma. Self-supervised Learning to Visually Detect the Terrain Surface for Autonomous Robots Operating in Forested Terrain, (03 2011)
03/11/2011	4.00	S. Zhou, J. Gong, G. Xiong, H. Chen, K. Iagnemma. Road Detection Using Support Vector Machine based on Online Learning and Evaluation, (03 2011)
03/11/2011	5.00	M. McDaniel, T. Nishihata, C. Brooks, P. Salesses, K. Iagnemma. Modeling Tree Stems in Forested Environments Using Ground-Based LIDAR, (03 2011)

TOTAL: 5

Number of Manuscripts:

Books

Received Paper

TOTAL:

Patents Submitted

Patents Awarded

Awards

Graduate Students

<u>NAME</u>	<u>PERCENT SUPPORTED</u>
FTE Equivalent:	
Total Number:	

Names of Post Doctorates

<u>NAME</u>	<u>PERCENT SUPPORTED</u>
FTE Equivalent:	
Total Number:	

Names of Faculty Supported

<u>NAME</u>	<u>PERCENT SUPPORTED</u>	National Academy Member
Karl Iagnemma	0.20	
FTE Equivalent:	0.20	
Total Number:	1	

Names of Under Graduate students supported

<u>NAME</u>	<u>PERCENT SUPPORTED</u>
FTE Equivalent:	
Total Number:	

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: 0.00

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:..... 0.00

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):..... 0.00

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense 0.00

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields: 0.00

Names of Personnel receiving masters degrees

<u>NAME</u>
Total Number:

Names of personnel receiving PhDs

<u>NAME</u>

Total Number:

Names of other research staff

<u>NAME</u>

<u>PERCENT SUPPORTED</u>

FTE Equivalent:

Total Number:

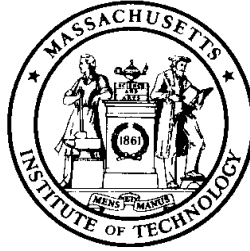
Sub Contractors (DD882)

Inventions (DD882)

Scientific Progress

See Attachment.

Technology Transfer



SELF-SUPERVISED MOBILITY-BASED TERRAIN CLASSIFICATION FOR UNMANNED GROUND VEHICLES

ARO Award Number: W911NF-07-1-0540

Final Report
08/24/2007 to 04/23/2013

Prepared for:

U.S. Army Research Office
Terrestrial Sciences Division
P.O. Box 12211
Research Triangle Park, NC 27709-2211

Attention:
Program Manager, Terrestrial Sciences Program

Technical POC:
Dr. Karl Iagnemma
Department of Mechanical Engineering
Massachusetts Institute of Technology
77 Massachusetts Avenue, Room 35-237a
Cambridge, MA 02139
Tel #: 617-452-3262 Fax #: 617-258-5802
Email: kdi@mit.edu

Abstract

Unmanned ground vehicles (UGVs) must rapidly and robustly characterize the nature of the terrain they are traversing, to improve autonomous mobility. This research program has focused on the development of a framework for self-supervised terrain classification, which allows a UGV to automatically learn the properties of terrain without human guidance. Work has also focused on novel applications of the self-supervised terrain learning approach, including urban/semi-urban driving on road networks. Finally, research has led to the development of novel sensing techniques for analyzing robot-terrain interaction mechanics at the micro scale. This report summarizes advances in all of these research areas.

1 Introduction—Self-Supervised Terrain Classification

This section describes a self-supervised learning framework that will enable a robotic system to learn to predict mechanical properties of distant terrain, based on measurements of mechanical properties of similar terrain that has been previously traversed. In this framework, a proprioceptive terrain classifier is used to distinguish terrain classes based on features derived from rover-terrain interaction, and labels from this classifier are used to train an exteroceptive (i.e. vision-based) terrain classifier. Once trained, the vision-based classifier is able to recognize similar terrain classes in stereo imagery. This section presents two distinct proprioceptive classifiers—a novel approach based on optimization of a traction force model and a previously described approach based on wheel vibration—as well as a vision-based terrain classification approach suitable for environments with unexpected appearance. The high accuracy of the self-supervised learning framework and its supporting algorithms is demonstrated using experimental data from a four-wheeled robot in an outdoor, Mars-analog environment.

The ability for humans to explore the surface of other planets using mobile robots (“rovers”) is fundamentally dependent on the autonomous mobility capabilities of these robots. Because targets of scientific interest such as craters, ravines, and cliffs present dangers to landing, planetary rovers must land at safe locations and travel long distances to reach these targets (NASA/JPL, 2007). Close teleoperational supervision of robots is not desirable because limited communication with operators on Earth places significant restrictions on the distance a rover can travel during a mission lifetime—for each downlink/uplink cycle of roughly 24 hours (Mishkin & Laubach, 2006), the rover cannot safely travel beyond the distance it can image with its cameras, which has been as little as 15 meters or less in dune fields observed by the Mars Exploration Rovers (NASA/JPL, 2005). Thus, advances in robot autonomy will lead to payoffs in terms of scientific data return from locations that were previously unreachable, since it will allow rovers to travel longer distances with limited human supervision.

One current limitation to autonomous mobility is the inability of current rovers to autonomously identify terrain regions that can be safely traversed. Existing path planning algorithms can generate a route to a target that avoids known obstacles only if they are given an accurate map of the ease of traversability of the surrounding terrain (Goldberg, Maimone, & Matthies, 2002; Nilsson, 1982; Stentz, 1994). Unknown hazards have the potential to immobilize the rover, delaying or permanently preventing completion of the mission. Thus, autonomous navigation is generally restricted to environments that operators have previously determined to be relatively benign. The ability to autonomously detect possible hazards from a distance would enable safe autonomous travel in previously unexplored rough terrain.

While geometric¹ hazards, such as large rocks or cliffs, can be sensed remotely using range sensing techniques (Talukder et al., 2002), little research has addressed remote sensing of non-geometric hazards, such as loosely packed soil or sandy slopes. The importance of sensing non-geometric hazards was highlighted in April 2005, when the Mars Exploration Rover (MER)

¹ Here, geometric hazards are considered to be obstacles that prevent safe rover travel due primarily to their shape, and not to loss of traction between a wheel and the terrain. In contrast, non-geometric hazards are regions of terrain that are impassible due to their limited traction properties.

Opportunity became embedded in a dune of loosely packed drift material (Cowen, 2005). The terrain geometry was not hazardous, however the high compressibility of the loose drift material caused the wheels to sink deeply into the surface, and the combination the drift's low internal friction and the motion resistance due to sinkage prevented the rover from producing sufficient thrust to travel up the slope. Opportunity's progress was delayed for more than a month while engineers worked to extricate it. A similar embedding event experienced by the Spirit rover in 2010 lead to the end of its mobility operations (Grossman, 2010).

Since non-geometric hazards are highly dependent on wheel-terrain interaction properties, methods for characterizing such hazards have focused on measuring aspects of that interaction. Examples include wheel sinkage measurement (C. A. Brooks, Iagnemma, & Dubowsky, 2006; Wilcox, 1994), soil characterization (Iagnemma, Kang, Shibly, & Dubowsky, 2004), wheel slip detection (Reina, Ojeda, Milella, & Borenstein, 2006), and explicit traversability estimation (Kang, 2003). These methods rely on proprioceptive² terrain sensing, which characterizes only the terrain immediately under the rover wheels, and is thus of limited use for predictive hazard avoidance.

While planetary scientists have long employed exteroceptive sensors, such as cameras or LIDAR sensors, for terrain sensing (e.g. (Azimi-Sadjadi, Ghaloum, & Zoughi, 1993; Weszka, Dyer, & Rosenfeld, 1976), their work has often addressed sensing from satellites, and the effect of terrain on ground vehicle mobility has not been a primary concern. Recent research efforts such as the DARPA Grand Challenge (Iagnemma & Buehler, 2006) and DARPA LAGR program (Jackel, Krotkov, Perschbacher, Pippine, & C. Sullivan, 2006) have sparked interest in terrain sensing for ground vehicles. Autonomous training of vision-based classifiers has been demonstrated for road identification (Thrun et al., 2006), and other researchers have proposed similar autonomous learning approaches for differentiation of traversable and non-traversable terrain (Kim, Sun, Oh, Rehg, & Bobick, 2006), though their work has focused on the detection of geometric hazards rather than non-geometric hazards. Other researchers have used autonomous learning to estimate vehicle mobility in scenarios where the visual appearance of terrain classes are known a priori (Angelova, Matthies, Helmick, & Perona, 2007), implicitly assuming the rover will not encounter any unexpected terrain classes. More recently, researchers have used an approach similar to the one proposed here and in (C. A. Brooks, 2009) to learn to distinguish terrain classes defined using simple proprioceptively-sensed attributes (Krebs, Pradalier, & Siegwart, 2010), though the correspondence between the proprioceptively-sensed attributes and vehicle mobility is uncertain.

This report presents an approach to autonomously identifying potentially hazardous terrain from a distance, when the visual appearance of terrain is not known a priori. To accomplish this task, a self-supervised learning framework is employed, whereby the rover uses proprioceptive sensors to classify terrain it has traversed and learns to associate the visual appearance of the terrain with terrain class labels. Two proprioceptive classification approaches are presented, including a novel approach to grade terrain through optimization of a traction force model.

² Proprioceptive sensors measure the internal state of the rover, and therefore sense terrain through its interaction with the rover. In this work, wheel torque, wheel speed, and wheel sinkage are considered to be measured by proprioceptive sensors.

This report is divided into six sections. Section 1 is the introduction and describes related work. Section 2 presents an overview of the self-supervised classification framework. Sections 3 and 4 present classifiers that are used as components within the framework: proprioceptive classification is addressed in Section 3, and exteroceptive classification is addressed in Section 4. These components are combined in Section 5 to experimentally validate the full self-supervised classification framework. Conclusions are presented in Section 6.

2 Self-supervised classification framework

In the following, “self-supervised classification” refers to automatic training of a vision-based terrain classifier. Whereas in a traditional (i.e. manually) supervised classifier a human provides labeled training examples for each class of interest, in a self-supervised framework another classification algorithm identifies these training examples. In the context of this report, proprioceptive sensors are used to identify terrain patches associated with terrain classes of interest, and visual features associated with these terrain patches are used to train a vision-based classifier. This vision-based classifier then identifies instances of these terrain classes in distant scenes.

Self-supervised classification is a form of learning from experience. Figure 1 illustrates the accumulation of data used to train the vision-based classifier. Initially, the rover has no knowledge of the relationship between terrain appearance and terrain class. From its initial position, the rover extracts visual features from disjoint patches of surrounding terrain (Figure 1(a)). Figure 1(b) shows the rover after it has driven onto a patch of terrain for which it has acquired visual feature data. Using proprioceptive sensors (e.g. vibration sensors or torque sensors), the rover extracts features related to physical wheel-terrain interaction, then employs a classifier to label the terrain patch. Pairs of terrain class labels and associated visual features are stored in memory. When sufficient data is accumulated, a vision-based terrain classifier is trained, and class labels are associated with physical properties. This allows the rover to predict the physical properties of distant terrain (Figure 1(c)).

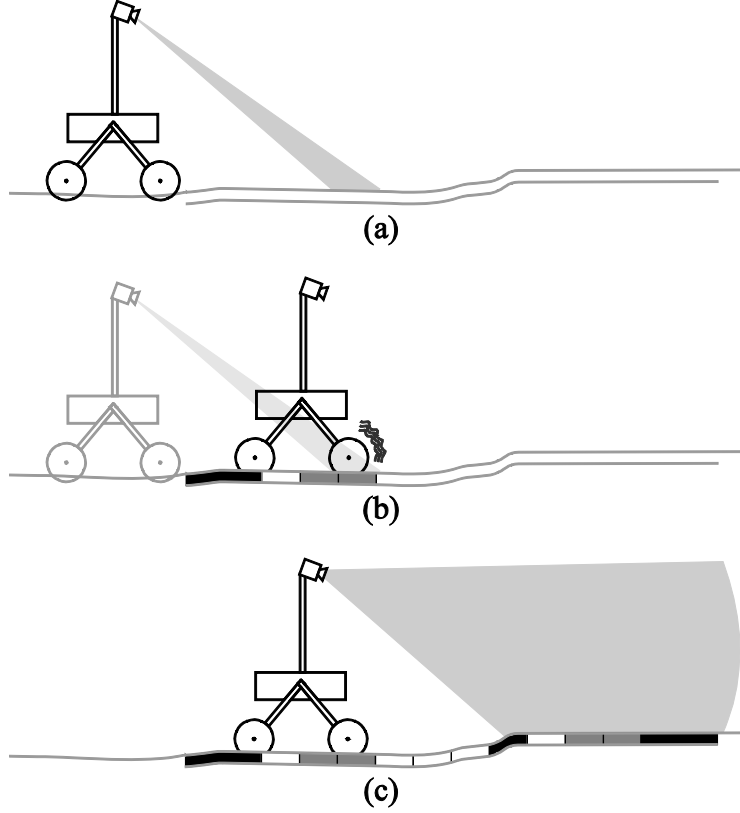


Figure 1: Illustration of proposed self-supervised classification framework

This framework has two distinct components: a proprioceptive terrain classifier and an exteroceptive terrain classifier. Figure 2 shows the information flow between these components. The proprioceptive classifier takes proprioceptive sensor data as an input and returns a terrain class label as its output. The exteroceptive terrain classifier takes exteroceptive sensor data (here, color stereo images of the terrain) as its input and returns terrain class labels for each terrain patch (here, 20 cm x 20 cm) in its field of view. These two classifiers are linked through the use of the proprioceptive classifier output as training labels for the exteroceptive classifier. The details of the proposed proprioceptive and exteroceptive classifiers are presented in Sections 3 and 4.

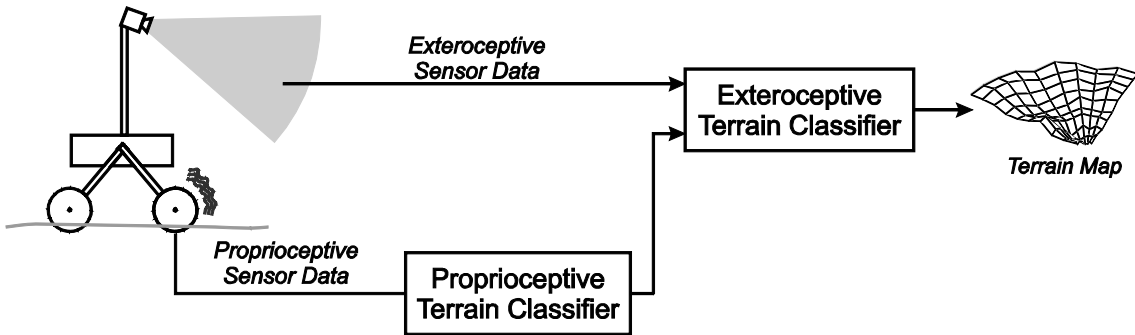


Figure 2: Information flow for self-supervised classification framework

3 Proprioceptive terrain classification

Because the proprioceptive classifier and exteroceptive classifier are linked only through the training approach, the underlying form of each classifier can be changed without affecting the other. Here we present two distinct proprioceptive classifiers, either of which can be used within the self-supervised framework. The first classifier, a vibration-based terrain classifier, uses traditional machine learning techniques to classify vibrations in the rover suspension arising from physical wheel-terrain interaction. This classifier requires a priori knowledge of the terrain classes in the environment, and corresponding hand-labeled training data. The second classifier, a novel traction-based terrain classifier, uses measurements of wheel torque and sinkage to estimate the minimum traction available at the wheel-terrain interface, and assigns a class label based on a set of pre-defined thresholds.

3.1 Vibration-based terrain classification

Vibration-based terrain classification is a method for classifying terrain patches based on vibrations induced in the rover structure by wheel-terrain interaction. Because mechanically distinct terrains induce distinct vibrations, features derived from these vibrations can be used as a means for classification. This approach relies on measurement of vibrations using an accelerometer mounted on the rover structure, representation of those vibrations in terms of the log-scaled power spectral density, and classification of the resulting features using a support vector machine (SVM) classifier. It is trained using hand-labeled vibration training data collected for each of the terrain classes during an offline learning phase.

The approach presented here for vibration-based terrain classification was initially developed in (C. Brooks, 2004) and (C. A. Brooks & Iagnemma, 2005). This report proposes an improved approach that employs an SVM classifier. This improved approach is validated using experimental data from a beach environment.

3.1.1 Approach

3.1.1.1 Description of vibration features

This algorithm represents each 1-second segment of vibration data as a vector of frequency-domain features. These features are calculated as follows. Given a time series of vibration signals $\mathbf{v}=[S_{vib,t=t_0}, \dots, S_{vib,t=t_0+1-1/F_s}]$ sampled at a frequency F_s , the first step is to compute the power spectral density (PSD) using Welch's method (Welch, 1967). Welch's method averages calculations of the power spectral density over eight subwindows to yield a 1025-element vector \mathbf{p} , where the i^{th} element, \mathbf{p}_i , is the estimate of the power spectral density at a frequency of $F_s(i-1)/2048$. Thus, \mathbf{p} is a time-shift-invariant representation of the vibration. To reduce the dominating effect of high-magnitude elements of \mathbf{p} , these magnitudes are log-scaled to yield a vector $\hat{\mathbf{p}}$.³

The vibration feature vector \mathbf{f} , is the set of elements from $\hat{\mathbf{p}}$ that correspond to a frequency range of interest between F_{min} and F_{max} . For this work, vibrations are sampled at 44.1 kHz, resulting in

³ This logarithmic scaling also has the advantage of representing time-domain convolution with vector addition. Thus, the log-scaled PSD of the convolution of two signals is equal to the sum of their log-scaled PSDs.

a spacing of 21.5 Hz between frequencies in the PSD estimate. The frequency range of interest is from 0 to 12 kHz. Thus, \mathbf{f} is a 558 element vector composed of the log-scaled PSD magnitudes associated with a single vibration segment.

3.1.1.2 Classifier description

To classify vibration features, an SVM classifier was implemented using the open-source library LIBSVM (Chang & C.-J. Lin, 2005, 2008). A Gaussian radial basis function (RBF) was used as the SVM kernel function, with parameters optimized by cross-validation over a set of vibration data not used for testing. (The optimized parameters were $C=100$ and $\gamma=5 \times 10^{-5}$.) The LIBSVM option to return predicted class likelihood was enabled.

During an offline training phase, the SVM was trained to recognize distinct terrain classes using vibration features calculated from traverses of the rover over terrain patches corresponding to each terrain class. In the online terrain classification process, vibration features associated with unlabeled terrain patches were calculated and these features were provided to the SVM for classification.

3.1.2 Experimental validation

The performance of the vibration-based terrain classifier was studied using data from experiments with the TORTOISE rover in an outdoor beach environment. This robot and environment will also be used to validate the exteroceptive classifier in Section 4, and the self-supervised classification framework in Section 5.

3.1.2.1 Robot configuration

TORTOISE, shown in Figure 3, is an 80-cm-long, 50-cm-wide, 90-cm tall robot with four 20-cm-diameter rigid aluminum wheels with grousers. The wheels on either side are connected to the main body and mast via a differential.



Figure 3: Photo of TORTOISE rover, showing location of wheel sensor suite

Mounted to the rover body is a two-axis tilt sensor (Crossbow CXTA02), measuring body pitch and roll. Additionally, all four wheel motors are equipped with encoders to measure wheel angular position. Wheel odometry, body pitch, and roll are used to align stereo-generated range data with an Earth-fixed reference frame.

TORTOISE measures vibration signals via a contact microphone mounted to the front right suspension strut of the rover, near the joint where the wheel axle passes through the strut, as seen in Figure 4. Vibration signals are recorded using the audio input of a laptop computer. During experiments, 16-bit samples of the vibration signal were collected at a frequency of 44.1kHz.

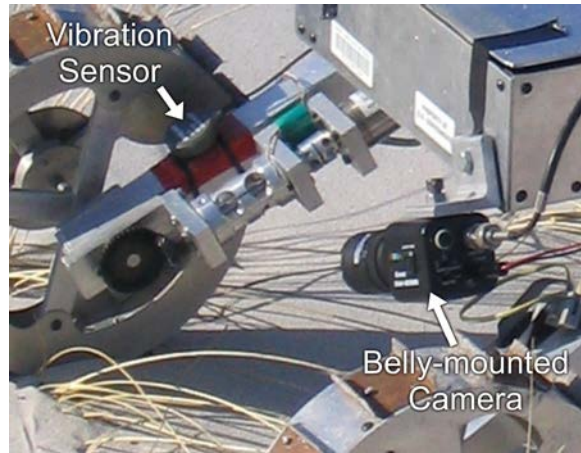


Figure 4: TORTOISE's wheel sensor suite, including vibration sensor and belly-mounted camera

TORTOISE is also equipped with a forward-looking stereo camera pair, which is mounted on a rigid mast 90 cm above the terrain. The stereo pair is a Videre Design “dual DCAM” with a 19 cm baseline and an overlapping field of view roughly $44^{\circ} \times 30^{\circ}$, capturing pairs of color images at 640×480 pixels each (Videre Design, 2001). Range data were extracted from the stereo images using SVS (Small Vision System), Videre Design's commercial stereo processing software (Konolige, 2007).

For these experiments, TORTOISE's belly-mounted camera, shown in Figure 4, captured images of the terrain being traversed. These images were used to allow a human to identify the terrain classes to serve as ground truth for classifier performance evaluation. A complete description of TORTOISE can be found in (C. A. Brooks, 2009).

3.1.2.2 Experiment environment

Experiments were performed at Wingaersheek Beach in Gloucester, Massachusetts, USA. This is a sandy beach with a mixture of small and large rock outcrops (relative to the size of the rover) as well as loose rocks. This site was chosen due to its similarity in appearance to the MER landing sites on Mars. In this environment, sand and rock were considered to be two distinct terrain classes. To demonstrate the ability of the classifier to work in a multi-class setting, matted piles of beach grass were used as a third terrain class. These three terrain classes are identified in Figure 5. Further details about the experiment environment can be found in (C. A. Brooks, 2009).



Figure 5: TORTOISE on Wingaersheek Beach, showing terrain classes

Three experimental data sets were collected, each during a rover traverse of at least 15 meters along a straight-line path containing a combination of the three terrains. No two paths were identical. During experiments, TORTOISE traveled at a speed of 3 cm/s. In all, 2283 seconds (38 minutes) of vibration data were collected.

3.1.2.3 Data processing

All vibration data was manually labeled to identify ground truth terrain classes underneath the front right wheel, based on the appearance of the terrain in images collected by the belly-mounted camera. Among all of the data sets, 1593 one-second vibration segments were labeled as sand (1289 segments), beach grass (209 segments), or rock (95 segments). Terrain under the other wheels was not recorded.

For the results presented here, cross-validation was used. Thus, each data set was used for testing the classifier that was generated using the remaining data sets as training data. Due to the reduced amount of training data, cross-validation is expected to under-predict the performance of a classifier generated using all three labeled data sets (Kohavi, 1995).

3.1.3 Results

The performance of the vibration-based terrain classifier was assessed by comparison to the hand-identified class labels (i.e., ground truth) using a receiver operating characteristic (ROC) curve, shown in Figure 6. In this plot, it can be seen that the classifier exhibits very good discrimination between each of the terrain classes. More than 50% of the terrain patches associated with rock are correctly identified before more than 1% of the non-rock terrain patches are incorrectly identified as rock. Similarly, 50% of the terrain patches associated with beach grass are correctly identified before 3% of the rock and sand terrain patches are falsely identified as beach grass. Classification of the sand class is also accurate, when a higher classification threshold is used, with 50% of terrain patches associated with sand correctly identified before 5% of the non-sand terrain patches are incorrectly identified as sand. Thus, combining all three

terrains, the vibration-based terrain classifier can classify 50% of the terrain patches while maintaining 92% confidence in the class label.⁴

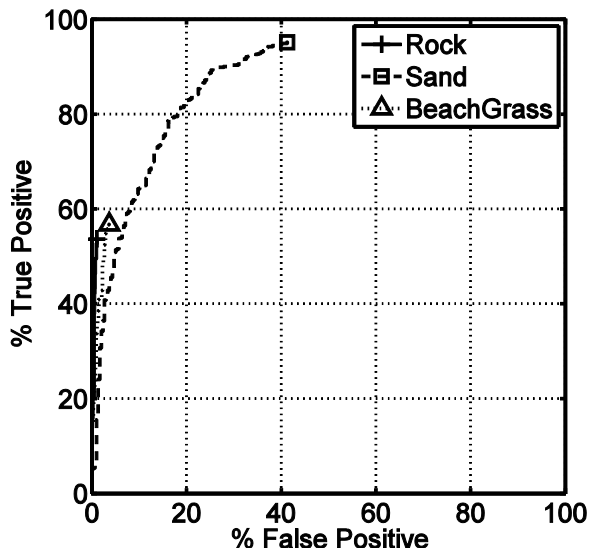


Figure 6: ROC curve for vibration-based terrain classifier

For comparison, note that random assignment of classes to terrains would yield equal values for true positive and false positive, resulting in a straight line from (0,0) towards (100,100). By definition, random assignment into three classes yields only a 33% confidence in the class label.

It should be noted that the low true positive detection rates for rock and beach grass with all terrain patches labeled—54% and 57%, respectively—reflects the fact that there were fewer examples of these terrain classes in the training data than there were for sand. This implicitly gives these two terrain classes a lower prior probability in the final classification. Thus, while they are correctly identified less often than sand, they have a correspondingly lower false positive rate. If a detection rate higher than that shown in Figure 6 is desired, more training examples can be provided, or a higher weight can be placed on the existing examples when training the SVM.

Because this vibration-based terrain classifier is used to label training data within the self-supervised classification framework, it is particularly important that the classification error rate be low so as not to corrupt the training of the vision-based classifier. Here, accuracy is improved by using a conservative classifier threshold and by combining multiple separate vibration-based class predictions for each terrain patch. (This is possible since the rover travels slowly.) If any of these terrain class assignments disagree, no training data from that terrain patch is used for training the vision-based classifier.

3.2 Traction-based terrain classification

An alternative proprioceptive terrain classification approach is based on estimating the maximum traction force available at the wheel-terrain interface, based on observed rover wheel torque and

⁴ 92.3% confidence is based on the observed mixture of terrains: 6% rock, 13% beach grass, and 81% sand.

sinkage. Class labels are then assigned class labels by comparing the estimated force to predetermined threshold levels. The analysis proposed here is intended to yield a single metric reflecting the ease of traversability of deformable terrain.

Previous researchers have taken various approaches to characterizing deformable terrain. Bekker, Wong, and Reece developed models for normal and shear stress acting on wheels in deformable terrain that can be used to calculate the net forces and torques on a wheel (Bekker, 1969; Wong, 2001; Wong & Reece, 1967). In their models, terrain is characterized by eight parameters. Measuring these parameters requires dedicated equipment to apply normal and shear forces and measure corresponding displacements. Iagnemma developed an approach to estimate the parameters of a reduced-order Bekker model without dedicated terrain sensing equipment, by measuring wheel torque and sinkage during a rover traverse (Iagnemma et al., 2004; Iagnemma, Shibly, & Dubowsky, 2002). Kang extended that work and proposed a nondimensionalized metric based on drawbar pull—the drag force that would be required to resist vehicle motion—as a traversability metric (Iagnemma, Kang, C. Brooks, & Dubowsky, 2003; Kang, 2003). This metric, known as the coefficient of traction (Wong, 2001), is calculated by dividing drawbar pull by the vertical load, and it represents the available net traction force as a fraction of the weight on a wheel. Kang found an approximate equation for the coefficient of traction as a function of wheel sinkage, wheel torque and vertical load. However, while Kang’s predictions of the coefficient of traction accurately approximate the predictions of the Bekker model when averaged over many terrains, he provided no guarantees about the error of any single prediction. For planetary exploration applications, overly optimistic predictions related to the traversability of terrain can lead to catastrophic failure.

This section presents a novel, optimization-based method for predicting strict upper and lower bounds on the coefficient of traction for a terrain patch. By assigning classes based on the lower bound on this traversability metric, this approach serves as a method for classifying terrain traversability in potentially high-risk scenarios.

3.2.1 Approach

In this section we describe (1) the coefficient of traction to be used as a traversability metric, (2) the wheel-terrain interaction model, and (3) the proposed optimization method to find upper and lower bounds on the coefficient of traction for a given patch of terrain. The mapping from traction force bounds to terrain classes is described in Section 3.2.4.

3.2.1.1 Traversability metric

The traversability metric used here is the coefficient of traction, μ_{tr} , which is a measure of the net available traction force between the wheel and the terrain. The net traction force can be modeled via lumped forces acting on a single, rigid wheel, as shown in Figure 7. Here, W is the vertical load supported by the terrain (including the weight of the wheel), T is the torque exerted on the wheel by a drive motor, DP is the drawbar pull, and z is the wheel sinkage. Clearly, if the drawbar pull is positive, the wheel can exert a force to move the rover in the desired direction of travel. Conversely, if the drawbar pull is negative, resistance on the wheel will slow the rover, possibly causing the rover to become immobilized.

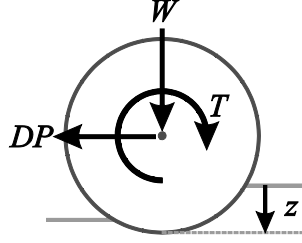


Figure 7: Wheel forces, torque, and sinkage

The coefficient of traction, μ_{tr} , is calculated as DP/W , and is related to the load a rover can tow relative to its own weight, as illustrated in Figure 8(a), or the maximum slope a rover can traverse, as illustrated in Figure 8(b). Neglecting redistribution of vertical loads on the wheels, the effect of slope on terrain internal stresses, and changes in μ_{tr} with the normal force (i.e. nonlinear wheel-terrain interaction effects), the wheel can travel up a slope of angle $\alpha = \text{atan}(\mu_{tr})$.

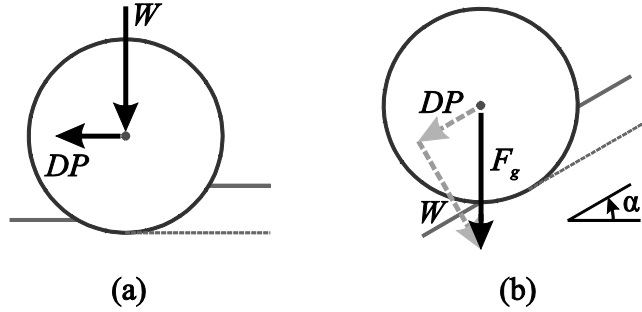


Figure 8: Wheel forces on flat terrain (a) and slopes (b)

It is important to note that the drawbar pull is a function of both the mechanical properties of terrain being traversed and the wheel slip ratio, i , which is defined as

$$i = 1 - \frac{v_x}{\omega r}, \quad (1)$$

where v_x is the forward velocity of the wheel, ω is the angular velocity, and r is the wheel radius. The relationship between drawbar pull and wheel slip ratio is illustrated in Figure 9, which shows experimentally observed relationships for four of the terrains studied later in this report. Note that for some of the terrains the drawbar pull is negative, indicating that an external force opposite DP is required to maintain a constant velocity at the specified slip ratio. Since the traversability metric is a function of drawbar pull, the value of wheel slip must be specified for the traversability metric to be measured on a given terrain. Here, the drawbar pull is measured for a wheel slip ratio between 0.4 and 0.7, conditions under which the drawbar pull is relatively insensitive to changes in slip for many terrains.⁵

⁵ If a rover is traveling with a slip ratio less than 0.4, and the drawbar pull it can exert is insufficient to maintain its forward progress, the slip ratio will tend to increase. Thus, at some point the rover is likely to be able to exert the drawbar pull calculated for higher-slip conditions.

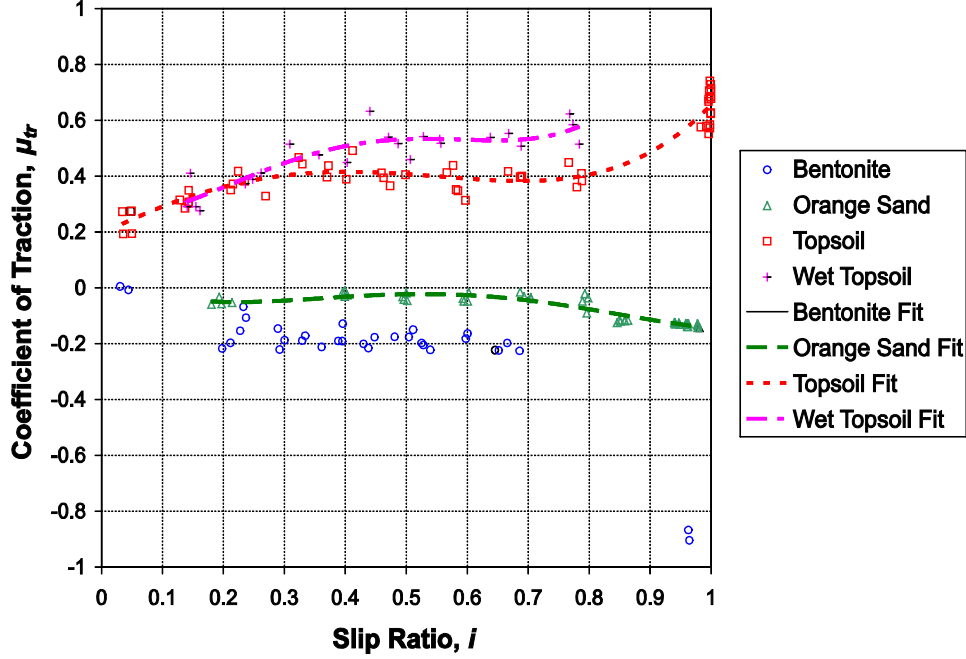


Figure 9: The coefficient of traction as a function of slip for four terrains (experimental data points and best fit curves)

3.2.1.2 Terrain sensing

In this work, it is assumed that the rover can measure the torque and absolute sinkage on (at least) one driven wheel. Torque can be measured directly, using a torque sensor, or it can be estimated based on wheel motor current. Absolute wheel sinkage—the distance between the undisturbed soil surface and the lowest point on the wheel—can be measured using a camera with a view of the side of the wheel (C. A. Brooks et al., 2006; Reina et al., 2006) or by a dedicated sinkage sensor.

3.2.1.3 Terrain model

A terrain model is employed to relate observed wheel sinkage and torque to predicted coefficient of traction. Here we use a classical Bekker model (Bekker, 1969), which defines parametric functions for the normal stress $\sigma(\theta)$ and tangential stress $\tau(\theta)$ along the rim of the wheel, as shown in Figure 10.

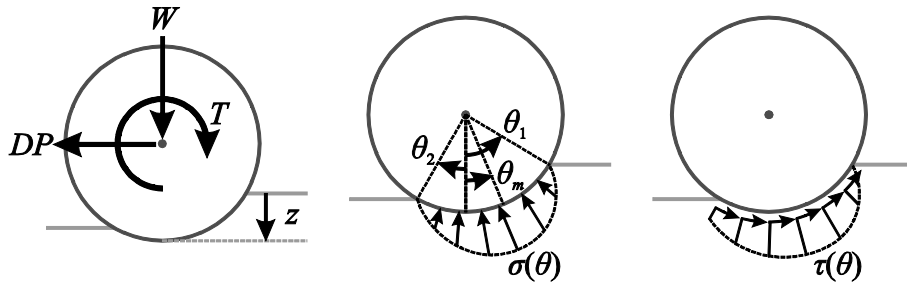


Figure 10: Bekker terrain model

The equation describing the normal stress function is as follows:

$$\sigma(\theta) = \left(\frac{k_c}{b} + k_\phi \right) \left(r \cos \theta^*(\theta) - r \cos \theta_1 \right)^n, \quad (2)$$

using

$$\theta^*(\theta) = \begin{cases} \theta & \text{for } \theta \geq \theta_m \\ \theta_1 - \frac{\theta_1 - \theta_m}{\theta_m - \theta_2} (\theta - \theta_2) & \text{for } \theta \leq \theta_m \end{cases}, \quad (3)$$

$$\theta_m = (c_1 + c_2 i) \theta_1, \quad (4)$$

and the equation describing the shear stress equation is as follows:

$$\tau(\theta) = (c + \sigma(\theta) \tan \phi) \left(1 - \exp \left(\frac{-j(\theta)}{K} \right) \right), \quad (5)$$

using

$$j(\theta) = r(\theta_1 - \theta - (1 - i)(\sin \theta_1 - \sin \theta)). \quad (6)$$

where i is the slip ratio from (1).

To compute the net forces applied to the wheel by the terrain, these stresses are integrated over the contact region:

$$W = rb \int_{\theta_2}^{\theta_1} \sigma(\theta) \cos \theta d\theta + rb \int_{\theta_2}^{\theta_1} \tau(\theta) \sin \theta d\theta \quad (7)$$

$$DP = rb \int_{\theta_2}^{\theta_1} \tau(\theta) \cos \theta d\theta - rb \int_{\theta_2}^{\theta_1} \sigma(\theta) \sin \theta d\theta \quad (8)$$

$$T = r^2 b \int_{\theta_2}^{\theta_1} \tau(\theta) d\theta. \quad (9)$$

Here, b is the wheel width. Since the wheel is assumed to be in equilibrium, W is the vertical load on the wheel, DP is the drawbar pull, and T is the torque applied by the motor. Using (7) and (8), the coefficient of traction μ_{tr} can be calculated from the normal and tangential stresses.

In this model, a terrain is characterized by the following Bekker parameters: k_c , k_ϕ , n , c_1 , c_2 , c , ϕ , and K , defined in Table 1.⁶ These parameters have been measured for a wide range of terrains (Wong, 2001), including (for some parameters) for Lunar and Mars regolith (Moore, Hutton,

⁶ It should be noted that while the foundations of this model were introduced by Bekker, Equation (6) for θ_m is due to Wong and Reece (1967). An alternative parameterization of (4) by Wong and Reece includes an additional factor of $b^{(1-n)}$, and replaces k_c and k_ϕ with $c k_c'$ and $\gamma_s k_\phi'$.

Scott, Spitzer, & Shorthill, 1977), (Rover Team, 1997), (Arvidson et al., 2004). By using the minimum and maximum values for each parameter observed in published data, plausible ranges for each of these parameters can be defined for a wide range of soil types. Table 2 lists the parameter ranges assumed for this Bekker model.

Table 1: Bekker model parameters

Symbol	Name	Role
k_c, k_ϕ	sinkage moduli	k_c and k_ϕ determine the magnitude of normal stress as a function of vertical soil deflection
n	sinkage exponent	n determines the rate of change of normal stress as a function of vertical soil deflection
c_1, c_2	wheel slip coefficients	c_1 and c_2 determine the location of the maximum normal stress as a function of wheel slip
c	soil cohesion	c determines the maximum shear stress which can be supported at zero normal stress
ϕ	angle of internal friction	ϕ determines the ratio between shear stress and normal stress
K	shear deformation modulus	K determines the rate of change of shear stress as a function of soil deformation

Table 2: Ranges for Bekker parameters ⁷

Parameter	Minimum Value	Maximum Value	Units
$k_c/b + k_\phi$	1000	3000	kPa/m ⁿ
n	0.578	1.2	
c_1	0.18	0.43	
c_2	0.32	0.41	
c	0	42	kPa
ϕ	15	43	deg
K	0.01	0.04	m

3.2.1.4 Optimization framework description

An optimization framework is used to find bounds on the value of the traversability metric, μ_{tr} , given a set of experimental observations and constraints on model parameter values. Specifically, to find a lower bound on the traversability metric, μ_{tr} is minimized subject to the constraints on the model parameters, experimentally observed torque T , sinkage z , slip i , and vertical load W .

Due to the complexity of the stress functions in the Bekker model, derivation of an explicit solution for μ_{tr} bounds is not feasible, so numerical optimization is employed. Constrained optimization is implemented using sequential quadratic programming (SQP) using the Matlab fmincon function (*Matlab (Version 7.1) with Optimization Toolbox (Version 3.0.3)*, 2005). In

⁷ The value for $k_c/b + k_\phi$ is given as a single parameter, since k_c and k_ϕ cannot be independently identified with a single wheel width. This range assumes a wheel width of 0.051 meters, the width of the TORTOISE rover's wheels.

this optimization, μ_{tr} , calculated using (7) and (8), is minimized (or maximized) over the parameters defined in Table 2. The optimization problem to find the lower bound is posed as the following minimization:

$$\begin{aligned}
(\mu_{tr})_{\min}^* &= \min_{(k_c/b+k_\phi), n, c_1, c_2, c, \tan \varphi, K, \theta_2} \frac{rb \int_{\theta_2}^{\theta_1} \tau(\theta) \cos \theta d\theta - rb \int_{\theta_2}^{\theta_1} \sigma(\theta) \sin \theta d\theta}{W_{measured}} \\
\text{subject to } T_{measured} &= r^2 b \int_{\theta_2}^{\theta_1} \tau(\theta) d\theta \\
W_{measured} &= rb \int_{\theta_2}^{\theta_1} \sigma(\theta) \cos \theta d\theta + rb \int_{\theta_2}^{\theta_1} \tau(\theta) \sin \theta d\theta \\
1000 &\leq (k_c / b + k_\phi) \leq 3000 \\
0.578 &\leq n \leq 1.2 \\
0.18 &\leq c_1 \leq 0.43 \\
0.32 &\leq c_2 \leq 0.41 \\
0 &\leq c \leq 42 \\
\tan 15^\circ &\leq \tan \varphi \leq \tan 43^\circ \\
0.01 &\leq K \leq 0.04 \\
-\theta_1 &\leq \theta_2 \leq 0,
\end{aligned} \tag{10}$$

where $\sigma(\theta)$ is the normal stress distribution calculated using (2), and $\tau(\theta)$ is the shear stress distribution calculated using (5). The optimization problem to find the upper bound is a maximization with the same arguments and bounds. Here the experimentally observed torque $T_{measured}$ and vertical load $W_{measured}$ are enforced as equality constraints in the optimization. Sinkage angle θ_1 can be calculated directly from z_{abs} , the absolute sinkage:

$$\theta_1 = \arccos \left(1 - \frac{z_{abs}}{r} \right). \tag{11}$$

The optimization routine was repeated ten times with randomly seeded initial parameter values. These parameters were passed through an initial optimization phase to find a set of parameters satisfying the equality constraints, prior to the μ_{tr} optimization.

3.2.2 Experimental validation

3.2.2.1 Wheel-terrain interaction testbed

Initial validation of the traction estimation algorithm was performed on data collected during experiments with a wheel-terrain interaction testbed, using five distinct terrains. These experiments were conducted by Kang for research presented in (Kang, 2003). The analysis of the data presented here represents work completed by the authors.

The wheel-terrain interaction testbed, shown in Figures 11 and 12, is designed to measure forces on a rigid wheel driven over terrain. It consists of a driven wheel mounted on an undriven vertical axis. The wheel-axis assembly is mounted on a driven carriage, so that the wheel forward velocity and angular velocity can be controlled independently. These testbed experiments were conducted using a rigid black plastic wheel 4.8 cm wide and 20 cm in diameter. Sand is bonded to the outside of the wheel to reduce slip at the wheel-terrain interface.

Drawbar pull is measured using a six-axis force-torque sensor mounted between the wheel assembly and the vertical axis. Wheel torque is measured using a rotating torque sensor mounted between the motor and the wheel. Wheel angular velocity is measured with a tachometer attached to the wheel motor, and the horizontal position of the carriage is measured with an encoder. Sinkage is measured using a linear potentiometer mounted on the carriage, not shown in the figure. The vertical load is adjusted by attaching steel plates to the top of the vertical axis.

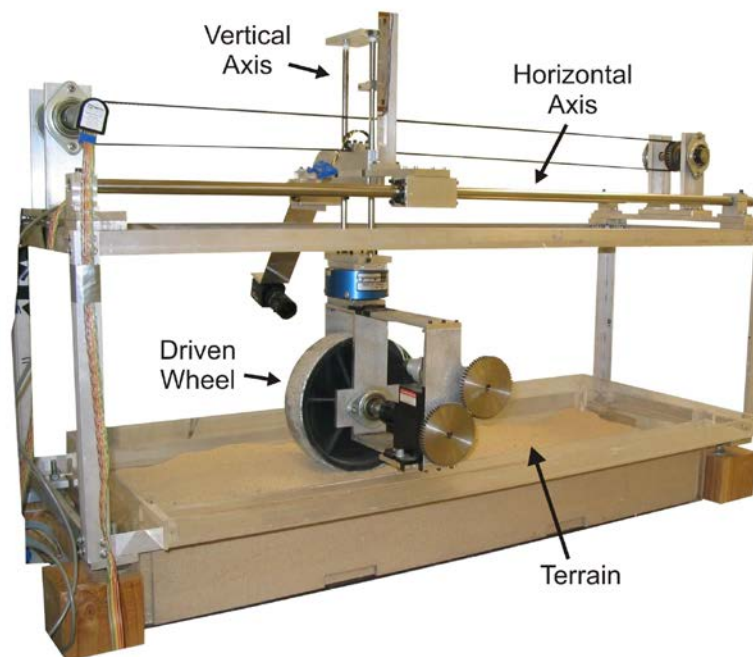


Figure 11: Wheel-terrain interaction testbed

The wheel travels in a 90-cm-long, 30-cm-wide, 15-cm-deep bin containing the terrain material. Five distinct terrains were used in these experiments: dry bentonite clay, modeling clay, orange sand, dry topsoil, and wet topsoil. The dry bentonite clay was a tan colored fine-grained material with the appearance of fine-grained sand. The modeling clay was a medium gray, damp, highly cohesive material that was flexible enough to be formed by hand, but rigid enough that it would maintain its shape once formed. The orange sand was fine-grained, nearly cohesionless dry sand. The topsoil was loamy black soil, either dry or saturated, as noted. (Moisture content can strongly affect certain soil physical properties.)

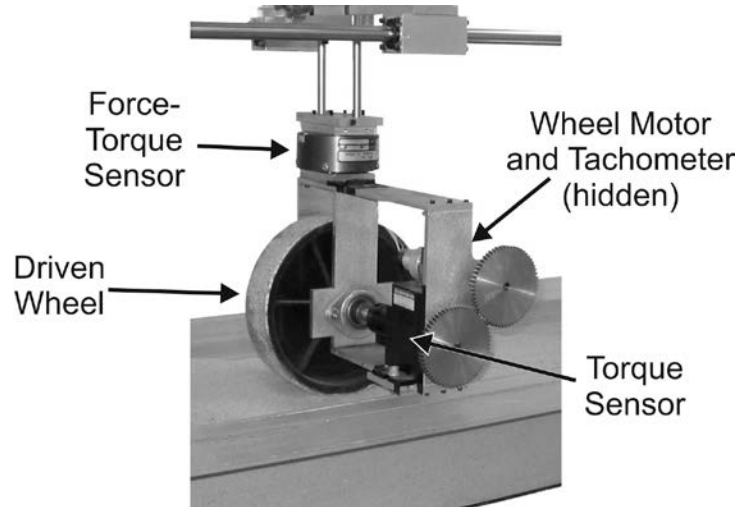


Figure 12: Wheel-terrain interaction testbed wheel with sensors

Each test run consisted of the wheel traveling from one end of the bin to the other at a specified forward velocity and angular velocity, over a single terrain and with a fixed vertical load. Twelve combinations of terrains and vertical loads were tested: bentonite at 21.4 N; clay at 53.2 N, 68.4 N, and 83.5 N; orange sand at 53.2 N, dry topsoil at 53.2 N, 60.8 N, 68.4 N, 76.0 N, and 83.5 N; and wet topsoil at 53.2 N and 68.4 N. Each of these combinations was run at least twice for each of two slip ratios, with the higher slip ratio ranging from 0.2 to 0.5. The lower slip ratio ranged from 0.04 to 0.23.

After the experiments were completed, steady-state wheel torque and sinkage values were extracted for each run, and the median value of torque and sinkage was used in the analysis for each combination of terrain, vertical load, and wheel slip.

3.2.2.2 TORTOISE experiments on Wingaersheek Beach

Additional experiments were conducted using the TORTOISE rover in an outdoor beach environment. For validation of the traction estimation approach, torque was measured using a torque sensor mounted to the motor driving the right front wheel, and wheel sinkage was measured using a camera mounted on the belly of the robot, with a view of the right front wheel, similar to the approach presented in (Brooks et al., 2006).

To induce wheel slip, the rover executed a pre-programmed behavior. In this behavior, the rover drove normally for 11 seconds, then (at $t=0$ seconds) spun the right-front wheel faster than the other three wheels. Since the rover body speed remained at (approximately) the speed of the other three wheels, the right-front wheel experienced a slip ratio of 33%. At $t=3$ seconds, the speed of the other three wheels was reduced, reducing the rover body speed and increasing the slip ratio to 50%. At $t=6$ seconds, the speed of the other three wheels was further reduced, increasing the slip ratio to 67%. At $t=9$ seconds, the rover resumed normal driving. The process was repeated over the full length of a traverse.

For this work, analysis was performed using wheel torque recorded during the second half of the 50% slip stage (i.e. from $t=4.5$ seconds to $t=6$ seconds). The 50% slip stage was selected because

50% slip is close to the range of slip ratios studied on the wheel-terrain interaction testbed. The second half of the stage was used to allow the wheel sinkage and torque to reach their steady state values.

Data from a single rover traverse of 11 meters over all three terrain classes was collected. Absolute wheel sinkage was measured using the images from the belly-mounted camera by hand-labeling the wheel rim and wheel-terrain interface. Images from the belly-mounted camera were also used to identify the terrain being traversed at any instant for ground truth purposes. Wheel sinkage and torque were passed to the optimization algorithm to determine bounds on the coefficient of traction, μ_{tr} . No explicit measurement of the drawbar pull was made, so assessment of the accuracy of the bounds relies on consistency between the predicted bounds and the known terrain classes.

3.2.3 Results

3.2.3.1 Wheel-terrain interaction testbed

For each of the terrains, the experimentally observed drawbar pull was compared to the optimization-based predictions of upper and lower bounds. Figure 13 shows the results. In this figure, the horizontal axis indicates the terrain type and vertical load, and the vertical axis indicates the coefficient of traction, μ_{tr} . The triangles and squares indicate the predicted upper and lower bounds predicted for the coefficient of traction, and the circles indicate the experimentally observed coefficients of traction.

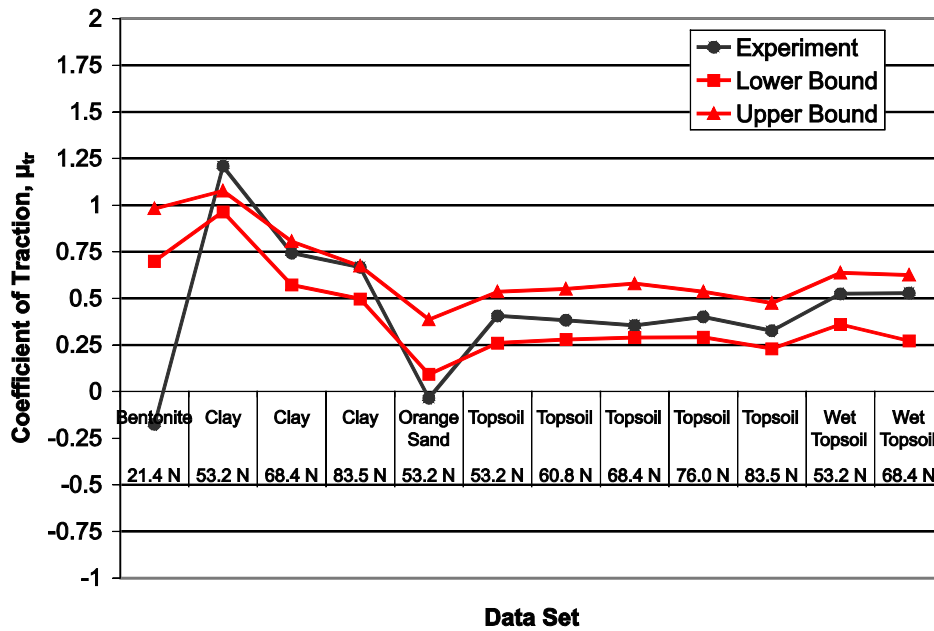


Figure 13: Testbed results for traction estimation

This approach shows relatively tight upper and lower bounds, with an average margin of 0.23 between the observed μ_{tr} and the upper bound, and an average margin of 0.21 between the observed μ_{tr} and the lower bound.

There are three cases in which the experimentally observed coefficient of traction lay outside the calculated bounds. For clay at 53.2 N the upper bound on μ_{tr} was too low, and for orange sand the lower bound was too high. This suggests that the model may not accurately represent the stress distributions in terrains producing very low (<0) or very high (>1) values of μ_{tr} . This suggests that the Bekker parameter ranges defined in Table 2 may need to be slightly widened to accurately model the terrains used in this experiment. Additionally, the lower bound for bentonite at 21.4 N was 0.87 higher than the experimentally observed μ_{tr} . This difference is large enough to suggest data collection errors specific to this data set, but since no errors were obvious in the data, this case was included for completeness. Without including the Bentonite data set, the average margin between the measured μ_{tr} and both the upper and lower bounds is 0.15.

3.2.3.2 TORTOISE rover on Wingersheek Beach

To study the performance of the traction estimation algorithm in an outdoor environment, the algorithm was applied to data from the TORTOISE rover on Wingersheek Beach, as shown in Figure 14. In this figure, the horizontal axis indicates the approximate position of the right-front wheel of the rover during the 50% spin state. The vertical axis indicates the coefficient of traction. The shade of the background indicates the terrain being traversed, as determined by manual labeling.

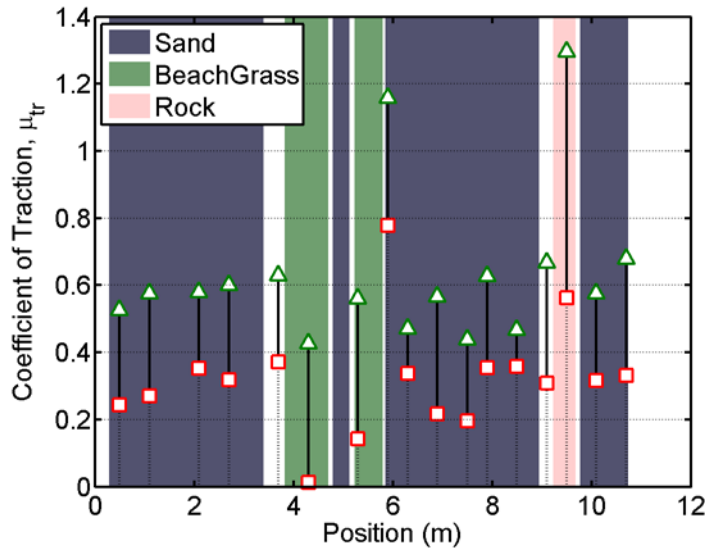


Figure 14: TORTOISE results for traction estimation algorithm

While no ground truth for the coefficient of traction is available for these results, and the Bekker equations are not expected to accurately model anisotropic terrain such as beach grass, they appear to be consistent with known physical characteristics of the terrains being traversed. In this chart, the μ_{tr} bounds appear to be lowest when the right-front wheel is on beach grass. This is consistent with the low available drawbar pull expected for beach grass, a highly compressible terrain that is relatively slippery. The predicted μ_{tr} bounds are highest when the wheel is on rock. This result is consistent with the high available drawbar pull expected for rough, highly cohesive rock. For sand, both the μ_{tr} bounds and the expected value of coefficient of traction lie between those of rock and beach grass.

3.2.4 Implementation of classification thresholds

To convert the quantitative traction bounds into discrete classes, as is needed by the self-supervised classification framework, pre-defined thresholds are used. Here, five terrain classes were defined based on the lower bound of μ_{tr} , to guarantee that on terrain of a given class the vehicle will be able to attain at least a specified traction force. These classes, labeled *A* through *E*, are shown in Table 3, and correspond to the ease of traversal of a terrain patch. Class *A* corresponds to terrain that can be easily traversed; class *E* corresponds to terrain that is untraversable. Classes *B* through *D* correspond to terrains that lie between those two extremes.

Table 3: Class labels and associated μ_{tr} ranges

Class Label	Range for μ_{tr} lower bound
<i>A</i>	0.5 to ∞
<i>B</i>	0.25 to 0.5
<i>C</i>	0.1 to 0.25
<i>D</i>	0 to 0.1
<i>E</i>	$-\infty$ to 0

4 Exteroceptive terrain classification

Exteroceptive terrain classification is the process of assigning class labels to terrain patches based on vision data collected from a rover’s cameras. Classification of terrain using visual features is an area that has received significant previous attention, from scientists studying land use (Olsen, Garner, & Van Dyke, 2002) to engineers designing navigation systems for autonomous robots (Rasmussen, 2002). Here, the vision-based terrain classification approach of Halatci is adopted (Halatci, 2006; Halatci, C. A. Brooks, & Iagnemma, 2008). This approach represents the appearance of a terrain patch via color, visual texture, and geometric feature vectors. For each of these three sensing modes, a SVM classifier is used to estimate likelihoods of the terrain patch belonging to each of the known terrain classes. The three sensing modes are then combined using naïve Bayes fusion to estimate the combined class likelihoods. The terrain patch is classified as belonging to the terrain class with the highest likelihood.

4.1 Approach

The vision-based terrain classification method employed here operates by extracting visual features derived from color, visual texture, and range data. Separate SVM classifiers for each visual feature type are used to predict the likelihood that a particular terrain patch belongs to any given terrain class. The resulting class likelihoods are combined using naïve Bayes fusion to yield a combined class assignment.

4.1.1 Visual features

4.1.1.1 Color

Color data is directly available from the cameras as red, green, and blue (RGB) intensities. However, illumination intensity affects all three values in a raw RGB representation, which can lead to poor classification results. To reduce the effect of the overall illumination level, a

modified hue, saturation, and value (HSV) representation of color is used as in (Sofman et al., 2006). In this approach, hue (an angle) is represented as two values— $\sin(\text{hue})$ and $\cos(\text{hue})$ —to eliminate the artificial discontinuity at 2π . Thus, color is represented as a 4-element vector: $[\sin(\text{hue}), \cos(\text{hue}), \text{saturation}, \text{value}]$.

4.1.1.2 Visual texture

Visual texture is a measure of the local spatial variation in the intensity of an image. The approach proposed here uses a wavelet decomposition similar to the one demonstrated in (Espinal, Huntsberger, Jawerth, & Kubota, 1998). Here, a grayscale image is decomposed with the Haar wavelet (Strang, 1993). Three scales of wavelets are used (2, 4, and 8 pixels), each scale having horizontal, diagonal, and vertical wavelets, corresponding to estimating the derivative of the intensity in the horizontal, diagonal, and vertical directions at each length scale. Because this process is sensitive to local changes in intensity, the magnitudes of the wavelet coefficients are then averaged over windows of 11, 9, and 7 wavelets. Thus, visual texture is represented by a 9-element vector, composed of the window-averaged horizontal, diagonal, and vertical wavelet coefficients at each scale. A detailed explanation of the texture feature extraction approach can be found in (C. A. Brooks, 2009).

4.1.1.3 Geometry

Terrain geometry is available through stereo image processing. The raw output of a stereo processing algorithm is a cloud of range data points. Here the points are divided into a grid of 20-cm by 20-cm terrain patches projected onto a horizontal plane. Geometric features are statistics calculated from the elevation of points associated with each terrain patch.

The first element of the geometric feature vector is the average slope of the terrain, defined as the angle ϕ between the least-squares-fit plane and the horizontal plane, in radians. The second element is the mean-squared deviation of the points from the least-squares plane along its normal, σ_{\perp}^2 . This is the same as the minimum singular value of the points' covariance matrix. The third element is the variance in the height of the range data points, σ_z^2 . The fourth element is the height difference between the highest and lowest points within the patch, r_z . Thus, the geometry of each patch is represented as a 4-element vector: $[\phi, \sigma_{\perp}^2, \sigma_z^2, r_z]$.

4.1.2 Classifier description

The vision-based terrain classifier uses a support vector machine classifier, implemented using the open-source library LIBSVM (Chang & C.-J. Lin, 2005, 2008), as was used for the vibration-based terrain classifier. For vision-based classification, linear or low-order polynomial kernels are appropriate, to enable fast classification (see (C. A. Brooks, 2009) for details). Here, a linear kernel is used, with the cost factor C optimized by cross-validation over a subset of images used for training. (For this work the optimized value was $C=10$. The option to return class likelihoods was enabled.)

It has been previously demonstrated that a straightforward approach of concatenating the color, visual texture, and geometric features into a single feature vector can yield poor classification results (Halatci et al., 2008), so a naïve Bayes fusion approach is used here. This approach assumes that color, visual texture, and geometric features are conditionally independent given the

terrain class. Note that since there may be many pixels observed in each terrain patch, the overall estimate of the class likelihood, based on the pixels' color or texture data, is taken as the geometric mean of the class likelihoods of the individual pixels.

In this supervised classification framework, hand-labeled feature vectors associated with each of the classes are used for SVM training. In the naïve Bayes fusion approach, separate SVM models are trained to classify color, visual texture, and geometry features. For this work 400 color feature vectors associated with each of the terrain classes is used to train the color SVM model. Visual texture and geometry SVM models are trained in the same manner.

4.2 Experimental validation

As with the local terrain classifiers, the vision-based terrain classifier was experimentally validated using data collected during experiments with the TORTOISE rover at Wingaersheek Beach. In these experiments, the same three terrain classes—sand, beach grass, and rock—were identified. To the rover's stereo camera, described in Section 3.1.2.1, sand appears as a uniform gray flat surface, rock appears tan and orange with some steep slopes and fine uniform texture, and beach grass appears highly textured with mixed browns and dark shadows.

Six experimental data sets were collected over the course of three days. Each data set consisted of a time series of stereo images and other sensor data recorded during a straight-line traverse of at least 10 meters over a combination of two or three terrains. No two paths were identical. During the experiments lighting conditions ranged from diffuse lighting from an overcast sky to harsh point lighting from low, direct sunlight. In all, 1646 image pairs were collected along with corresponding internal sensor data.

The stored data collected during the experiments was post-processed offline. Every 20th image pair was hand-labeled to identify the ground-truth terrain class corresponding to each pixel. For each of these labeled image pairs, range data was also calculated using SVS (Konolige, 2007), discarding potentially unreliable data beyond 8 meters. By combining the labels with the range data, ground-truth terrain classes were identified for each 20-cm by 20-cm terrain patch. For each of the six data sets, between 10 and 27 image pairs were hand labeled. The first two or three image pairs from each data set were used for training the classifiers, with the remaining images used for testing. Note that separate classifiers were trained and tested for each data set.

4.3 Results

The accuracy of the vision-based terrain classifier was assessed for each of the data sets. Figure 15 shows the receiver operating characteristic (ROC) curves for a representative data set, created by varying the confidence required for classification. Note that the scale of the x-axis is magnified to allow the curves to be easily seen.

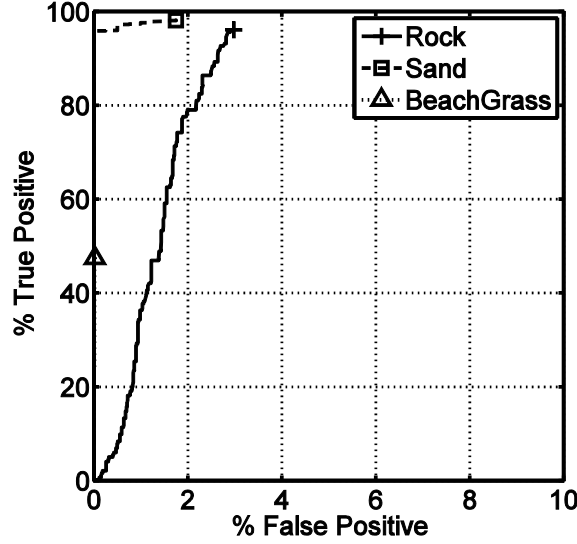


Figure 15: Representative ROC curves for vision-based terrain classifier

It can be seen that the manually trained classifier performed very well at identifying both sand and beach grass. More than 95% of the sand was correctly identified before any of the other terrains was falsely identified as sand. For beach grass, nearly 50% was correctly identified with less than 0.1% of the other terrains falsely identified. Results for rock were also very good, with 96% of the rock correctly identified and less than 3% of the other terrains falsely identified as rock.

Numerical results also indicate robust performance of the vision-based classifier across all six data sets, as shown in Table 4. The top two rows show statistics of the true positive percentage of the classifiers when no data is left unlabeled, corresponding to the vertical coordinate of the ROC curve endpoints. The third and fourth rows show statistics of the false positive percentage, corresponding to the horizontal coordinate of the ROC curve endpoints. The bottom two rows show statistics related to the ratio between the true positive percentage and the false positive percentage. The metric, $\%TP/(\%TP + \%FP)$, is closely related to the fraction of labeled patches which are labeled correctly. The values in brackets indicate a 95% confidence interval for the statistic.

It can be seen that on average more than 95% of each terrain class was correctly identified, with only 5% being falsely identified when all of the terrain patches were assigned a class. It should be noted, however, that the true positive rate and false positive rate tend to increase and decrease together, as more or less of the terrain is assigned a given class label. The metric presented in the bottom two rows is intended to measure accuracy while being insensitive to that variation. Here it can be seen that on average, 95% of terrain classified as a given terrain class actually belongs to that class, even when no terrain is left unclassified. This result shows that accurate vision-based classification can be accomplished using an SVM classifier with the proposed features in a natural outdoor environment. These vision-based classification results will be used for comparison in Section 5 to assess the performance of the self-supervised classification framework.

Table 4: Performance of vision-based terrain classifier

Mean % True Positive	95.1% [93.1% - 97.1%]
St. Dev. of % True Positive	1.9% [1.2% - 4.7%]
Mean % False Positive	4.9% [2.9% - 6.9%]
St. Dev. of % False Positive	1.9% [1.2% - 4.7%]
Mean % TP/(%TP + %FP)	0.95 [0.93 - 0.97]
St. Dev. of %TP/(%TP + %FP)	0.02 [0.01 - 0.05]

5 Self-supervised classification results

Traditional methods for sensing non-geometric hazards, such as wheel slip detection, rely on proprioceptive sensing of wheel-terrain interaction, and thus are limited to characterizing terrain patches in physical contact with the rover. To allow for *predictive* non-geometric hazard avoidance, remote detection of non-geometric hazards is needed. Self-supervised classification provides a method for generalizing information gained from proprioceptive sensors to yield information about distant terrain.

Here we present experimental results from two instantiations of the self-supervised classification framework introduced in Section 2. The first instantiation, presented in Section 5.1, uses a vibration-based terrain classifier to identify manually-labeled terrain classes. This instantiation is used to experimentally validate the self-supervised classification framework. However, because human supervision is required to train the vibration-based terrain classifier, this instantiation is not appropriate for implementation in environments where terrain properties are not known a priori.

The second instantiation, presented in Section 5.2, is designed for scenarios in which no a priori terrain knowledge is available. In this instantiation, terrain classes are defined to correspond to ranges of a traversability metric, and thus exteroceptive terrain class predictions are used to predict the traversability of distant terrain.

5.1 Self-supervised learning from vibration

In the self-supervised framework presented in this section, local terrain patches are classified based on the vibration signature in the rover structure caused by wheel terrain interaction, and distant terrain patches are classified based on stereo imagery. Visual data used for training is gathered from stored imagery from the stereo imagery.

5.1.1 Approach

A block diagram showing the inputs and outputs of the vision-based classifier is presented in Figure 16. In the training stage, shown in Figure 16(a), the vision-based classifier takes as inputs

(1) terrain class labels identified by the vibration-based terrain classifier, and (2) visual features associated with the labeled terrain patches. In the classification stage, shown in Figure 16(b) the vision-based classifier takes visual features as inputs, and outputs the terrain class associated with the observed terrain patches.

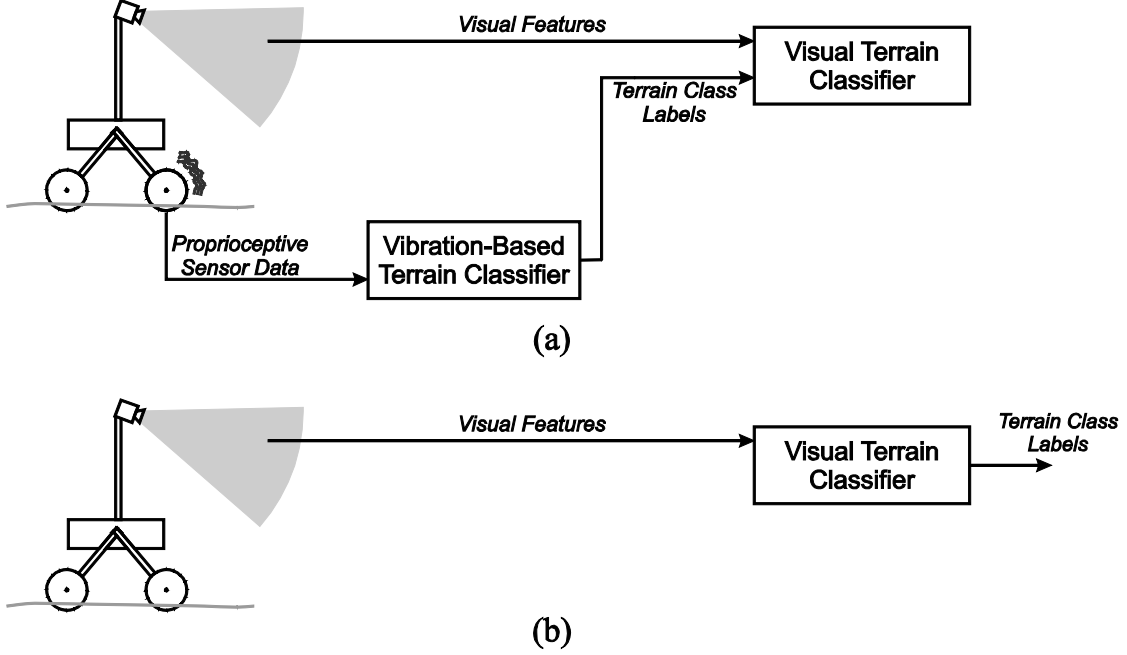


Figure 16: Information flow in self-supervised classification framework during (a) training and (b) classification

Here, the vibration-based terrain classifier, described in Section 3.1, takes the role of the “supervisory” classifier, since it supervises the labeling of training data. The “supervised” classifier is the vision-based terrain classifier described in Section 4. Training data for the vision-based classifier is extracted from forward-looking stereo images stored in memory and recalled when the rover classifies a previously observed terrain patch using proprioceptive sensors, as illustrated in Figure 17. Thus, color, visual texture, and geometry features (F_{color} , $F_{texture}$, and $F_{geometry}$) associated with a given terrain patch, located at position (x,y) are stored in memory after being observed by the forward-looking stereo cameras at time t_0 . When, at a later time t_1 , the rover reaches position (x,y) , it uses proprioceptive sensors to identify the terrain class C associated with terrain patch P . The proprioceptively identified terrain class C , and the remotely sensed visual features F_{color} , $F_{texture}$, and $F_{geometry}$ are then used to train the vision-based terrain classifier.

This approach relies on stereo processing to correlate image pixels with their corresponding terrain patch and on accurate position estimation to identify the location of the terrain patch the rover’s proprioceptive sensors are measuring. Here, odometry-based position estimation is used to identify where a proprioceptively sensed terrain patch appears in the stored images. This approach also assumes that most terrain patches contain only one terrain class, so that the class

sensed by the wheel passing over a 6-cm-wide band is representative of a full $20\text{ cm} \times 20\text{ cm}$ patch.

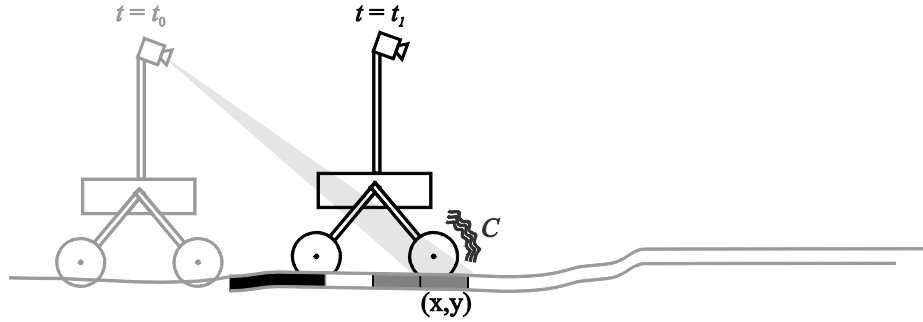


Figure 17: Illustration of visual information being associated with proprioceptively sensed labels

Training data for each terrain class is accumulated as the rover travels, and stored in memory. To limit the training time for the vision-based classifier, each terrain class is limited to a maximum of 400 sets of features. Older data is discarded if new data is collected that would exceed that maximum. Vision-based terrain classification is implemented on a patch level; each terrain patch is classified via naïve Bayes fusion of the color, visual texture, and geometric features sensed for pixels corresponding to that patch.

5.1.2 Experiment details

The accuracy of vision-based classification of terrain using the self-supervised training approach was compared to a traditional supervised vision-based classifier (presented in Section 4) using the same experimental data sets described in Section 4.2.

For the manually supervised classifier, the first two stereo image pairs with ground truth labels were used to train a vision-based classifier for each data set. Four hundred features for each sensing mode from each class were used for training the classifier. The remaining hand-labeled images from that data set were used for testing. Across all six data sets, 93 image pairs were used for assessing the accuracy of the vision-based classifiers.

For self-supervised classification, a separate vibration-based classifier was trained for each data set, using hand-labeled vibration data from the other data sets. This vibration-based classifier was then used to provide labels for the entire rover traverse. At the end of the traverse, the self-supervised vision-based classifier was trained using up to 400 features from each sensing mode, for each terrain class. The accuracy of the self-supervised classifier was tested using the same stereo test images as were used for testing the baseline manually supervised classifier.

5.1.3 Results

Figure 18 presents the ROC curves for the self-supervised terrain classifier. This plot shows the classification accuracy for one data set, based on classification of the same 26 images as Figure 15. Overall, self-supervised terrain classification for this data set is very good. More than 80% of both sand and beach grass are correctly classified before 1% of the other terrains are misclassified as either. While the classification of rock has a low false positive rate—less than

1.5%—the classifier is less successful at detecting rock than the other terrains. Approximately 25% of the rock was correctly classified for this data set. This low rate of correct classification of rock is similar to the behavior observed for the vibration classifier alone and may reflect the fact that there were fewer examples of rock along the vehicle’s path (and thus in the training data) than sand or beach grass.

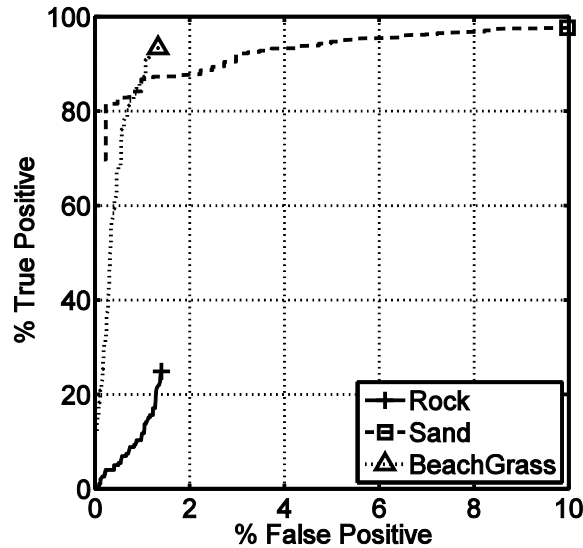


Figure 18: ROC curves for self-supervised classifier

Self-supervised classifiers and manually trained classifiers were implemented for each of the six data sets, and the results are shown in Table 5. Here, ROC curves were generated showing the combined true positive and false positive rates across all three terrain classes. The performance of each classifier on a data set was summarized by a single point on the ROC curves—the point at which the difference between the true positive percentage and the false positive percentage is at a maximum. The first two rows of the table show statistics of the true positive percentage of the classifiers, corresponding to the vertical coordinate of the optimal point along the ROC curves. The third and fourth rows show statistics of the false positive percentage, corresponding to the horizontal coordinate of the optimal point. The last two rows show statistics related to the ratio between the true positive percentage and the false positive percentage. This metric, $\%TP/(\%TP + \%FP)$, is the fraction of labeled patches which are labeled correctly. Note that the numbers in brackets indicate a 95% confidence interval for each metric.

In this table, it can be seen that the self-supervised classifier using remote training performs almost as well as the manually supervised classifier for each of the metrics. In fact, the difference in performance between the two classifiers is not statistically significant. The low values for the standard deviations suggest that this remote training approach to the self-supervised classification is robust.

The self-supervised approach is intended for situations when a manually trained classifier is not a viable option, due to the necessity of human labeling of terrain. In a planetary exploration setting, manual training would impose a significant delay between the time that training images were collected and the time that the trained classifier could be implemented. Thus, the accuracy

of a self-supervised classifier is more fairly compared to a manually trained classifier trained on a previously collected data set. In this scenario, the training images for the manually supervised classifier are hand-labeled images drawn from one data set, and the performance of the classifier is tested using images from the following data set, captured minutes or days later. This results in variation in illumination for some data sets. For example, one of the data sets was captured with overcast skies, and the next was captured with low-angle, direct sunlight.

Table 5: Comparison of self-supervised classification to manually supervised classification

	Self-Supervised Classifier using Remote Training	Manually Supervised Classifier	Manually Supervised Classifier (Prior Data Set)
Mean % True Positive	94.7% [91.0% - 98.3%]	94.2% [91.1% - 97.3%]	69.1% [29.7% - 100%]
St. Dev. of % True Positive	3.5% [2.2% - 8.5%]	2.9% [1.8% - 7.2%]	37.6% [23.4% - 92.1%]
Mean % False Positive	5.3% [1.5% - 9.1%]	3.8% [2.0% - 5.5%]	11.3% [0% - 24.6%]
St. Dev. of % False Positive	3.6% [2.2% - 8.8%]	1.7% [1.0% - 4.1%]	12.6% [7.9% - 31.0%]
Mean %TP/(%TP + %FP)	0.95 [0.92 - 0.99]	0.96 [0.94 - 0.98]	0.85 [0.66 - 1.0]
St. Dev. of %TP/(%TP + %FP)	0.03 [0.02 - 0.08]	0.02 [0.01 - 0.04]	0.16 [0.09 - 0.45]

The accuracy of such a classifier is presented in the third column of Table 5. Here the difference in performance between the self-supervised classification approach and the manually supervised classification approach is significant. The self-supervised classification approach yields better true positive classification, a lower false positive rate, and higher overall classification accuracy, as compared to the manually supervised classifier when training delay is added.

5.1.4 Computation time

To enable real time operation, an effort was made to limit the computational complexity of training and testing of these classifiers, so the most computationally intensive tasks were stereo data extraction and texture feature computation. Extraction of geometric features from a 3-D point cloud took an average of 5 seconds per image using a Matlab script on a Pentium 4 1.8 GHz desktop computer. Texture feature extraction took 17.3 sec, using an unoptimized Matlab script. A C-code implementation would be expected to run much faster.

Given pre-computed color, texture, and geometry features, training the vision-based classifier took 1.5 seconds on average using the LIBSVM library. Classification took an average of 4.1 seconds per image.

5.2 Self-supervised learning from mechanical characterization

For scenarios in which no a priori information about the terrain is available, a completely unsupervised terrain learning system can be assembled using the self-supervised classification framework and component algorithms developed and validated earlier in this report. As with the self-supervised learning system validated in Section 5.1, this terrain learning system learns to identify instances of a terrain class in the distance based on the appearance of proprioceptively sensed terrain patches. However, the terrain learning system presented in this section uses terrain classes defined based on an explicitly calculated traversability metric, eliminating the need for human supervision during the training of the proprioceptive terrain classifier. This unsupervised terrain learning system was applied to experimental data from the TORTOISE rover in natural outdoor terrain.

5.2.1 Approach

In this instantiation of the self-supervised learning framework, the traction-based terrain classifier described in Section 3.2 takes the role of the supervisory classifier. Here, for every local patch with associated wheel slip data, μ_{tr} bounds are calculated and a terrain class label is assigned based on the lower bound.

Vision-based terrain classification was implemented using a two-stage approach. In the first stage, a two-class SVM classifier estimates the likelihood of a terrain patch containing “novel” terrain (i.e. terrain not represented by training data) (C. A. Brooks, 2009; C. A. Brooks & Iagnemma, 2009). In the second stage, a separate multi-class SVM classifier identifies which of the known classes is most likely to be associated with the terrain. Both stages use the same color, texture, and geometry features to represent the terrain.

Since the two stages both output probabilities, it is straightforward to identify the probability of a distant terrain patch being associated with each of the five known classes (A , B , C , D , and E), or the unknown class, *Unknown*:

$$P(A) = P(A | \text{Known}) P(\text{Known}) \quad (12)$$

$$P(\text{Unknown}) = P(\text{Novel}), \quad (13)$$

where $P(\text{Known})$ and $P(\text{Novel})$ are outputs of the novel terrain detector, and $P(A|\text{Known})$ is one of the outputs of the known terrain classifier. Thus, the remote terrain classifier calculates the probability of each terrain cell belonging to each terrain class: $[P(A), P(B), P(C), P(D), P(E), P(\text{Unknown})]$.

Given the likelihood of each class, a traversability map can be produced, where the a single number represents the terrain traversability in each patch—a conservative estimate for the coefficient of traction, μ_{tr} . Here that conservative estimate of μ_{tr} is defined as the highest value for which there is at least an 80% probability that the true value would be higher.⁸ This can be calculated from the ranges of lower μ_{tr} bounds associated with the classes (Table 3) and the class probabilities from the remote terrain classifier. For example, given the probabilities $P(A) = 50\%$,

⁸ The value of 80% was chosen because it provides a reasonable balance between being too cautious (since the ranges are already estimates of the lower bound of DP/W) and being too optimistic (which could endanger the safety of a rover).

$P(B) = 25\%$, $P(C) = 6\%$, $P(D) = 10\%$, $P(E) = 4\%$, $P(Unknown) = 5\%$, a conservative estimate of μ_{tr} for the associated terrain cell would be 0.1 (i.e. the lower end of the range for class C), because $P(A) + P(B) + P(C) \geq 80\%$.

5.2.2 Experiment details

This instantiation of the self-supervised learning framework was applied to data from a single 10-meter traverse of TORTOISE on Wingaersheek Beach. This traverse contained regions of sand, beach grass, and rock. During the traverse, the rover performed the slip-inducing behavior described in Section 3.2.2.2, and all data was stored so that it could be passed to an offline implementation of the terrain classifier. It should be noted that, for the results presented here, wheel sinkage measurement was implemented in post-processing using a human to manually identify the wheel-terrain interface in images from the belly-mounted camera. This process could be automated using the visual wheel sinkage measurement approach presented in (C. A. Brooks et al., 2006).

5.2.3 Results

The output of the terrain learning system is a prediction of a lower bound of traction coefficient for each terrain patch as predicted using each stereo image. This is most easily viewed as a video, but still frames are shown in Figures 19-21.

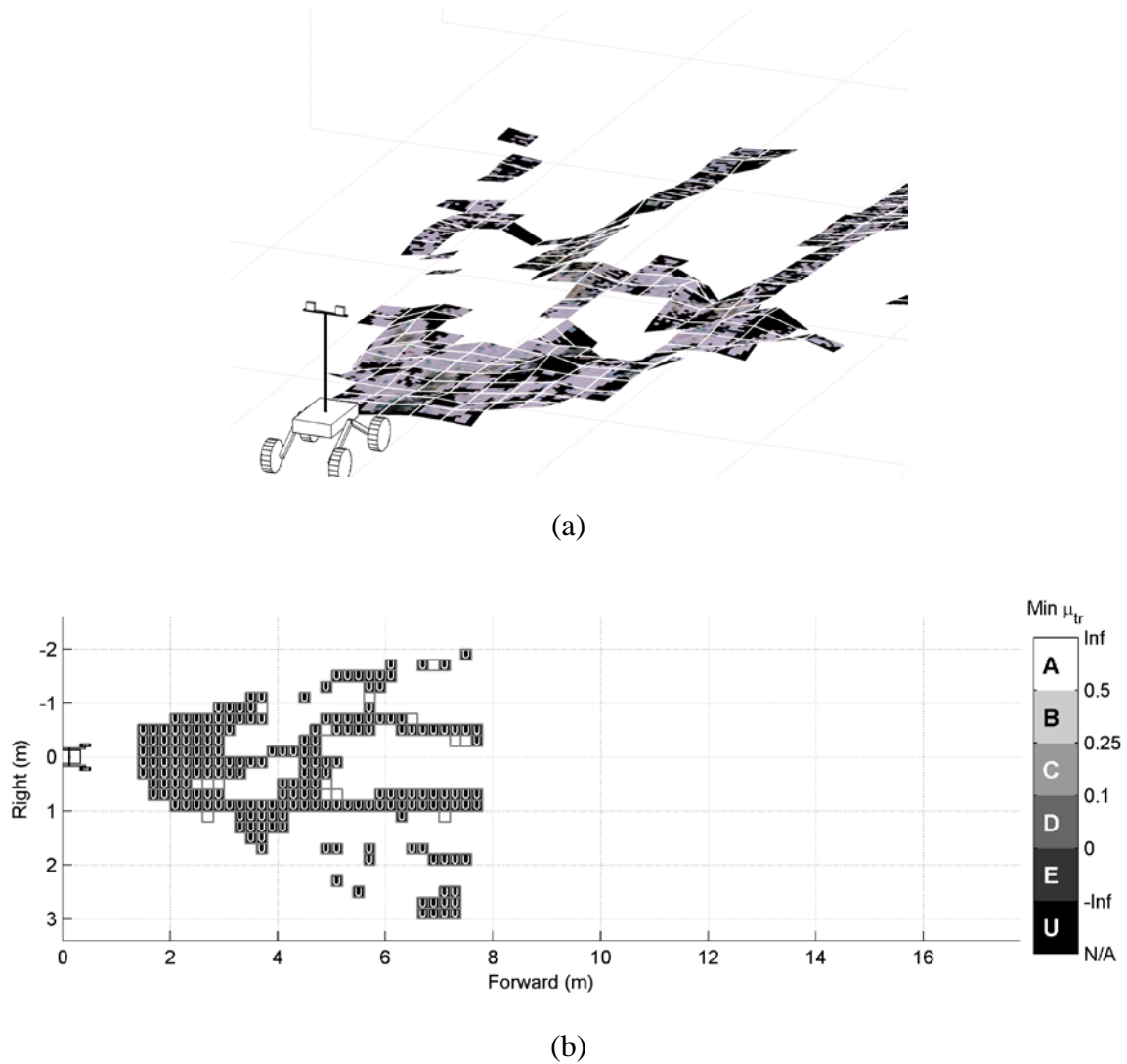


Figure 19: Terrain learning system results, at $t = 5.0$ sec, distance traveled = 0.13 m, (a) 3-D view, (b) plan view showing terrain classes

Figure 19 shows the rover's internal map of the terrain, just after it has started its traverse. Figure 19(a) shows a 3-D view illustrating the topography of the terrain as sensed by the rover (which is shown in the lower left corner of the image). At this instant, the range data calculated from the first two images from the stereo pair is sparse, as evidenced by the large gaps in the terrain map. Figure 19(b) shows a plan view of the terrain, with terrain patches labeled based on the predicted lower bound of the traction coefficient. Since the rover hasn't completed mechanical characterization of any of the terrain patches for which it has stereo data, there is no "known" terrain in the distance. All observed terrain is considered to be novel, and the terrain patches are labeled "U" (*Unknown*), signifying that it doesn't have sufficient experience to assess the traversability.

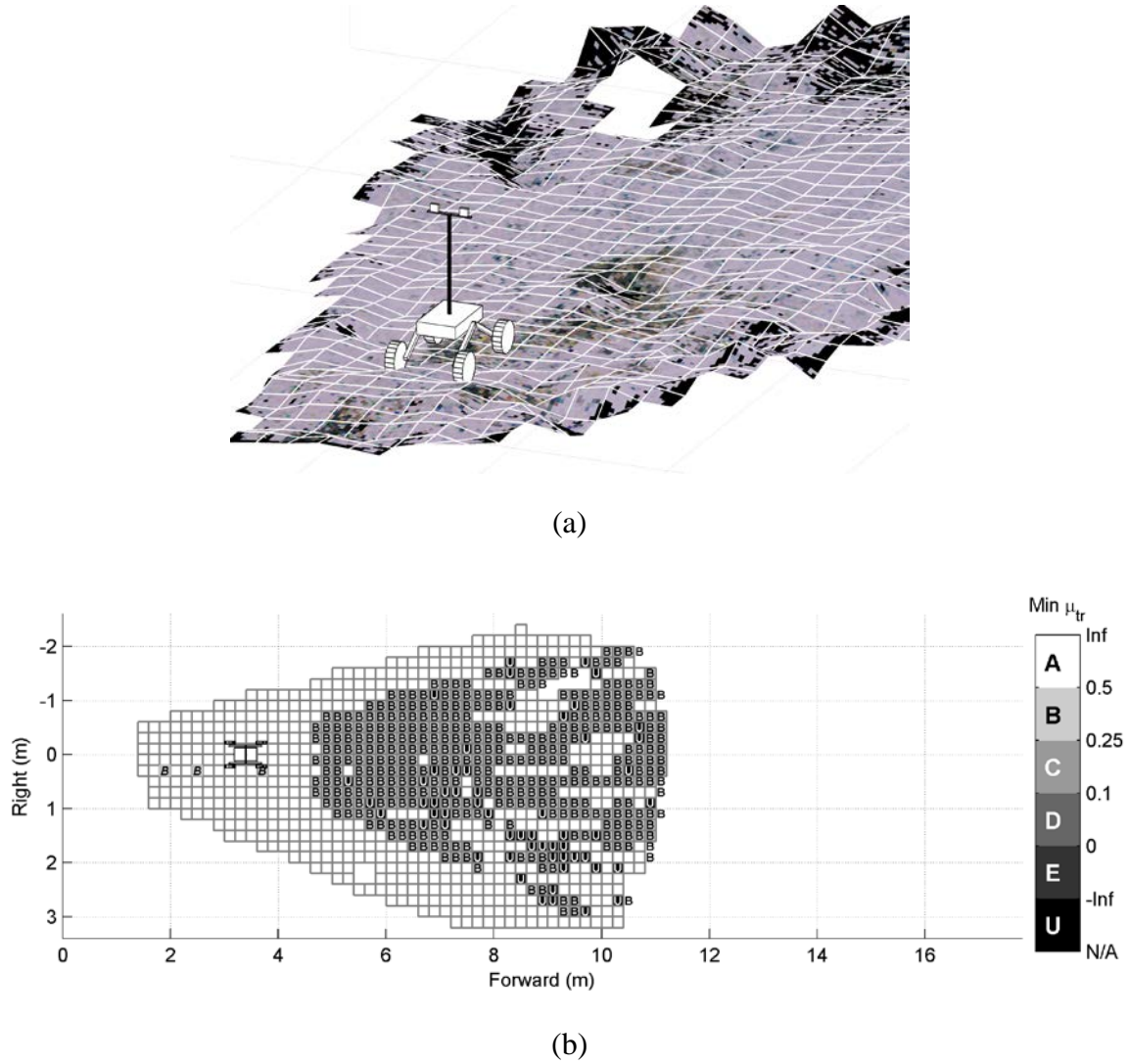
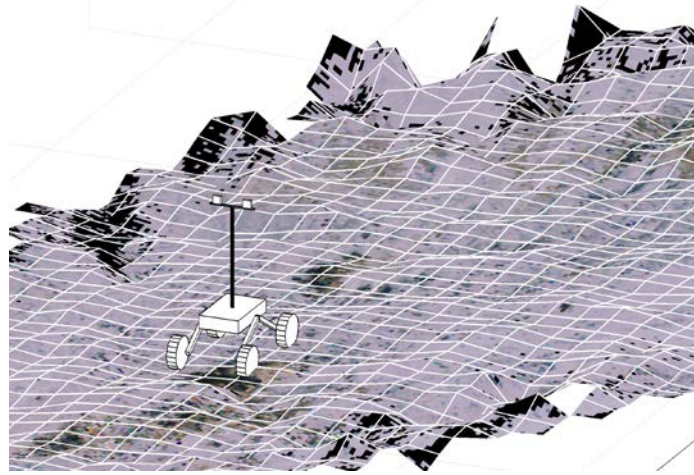
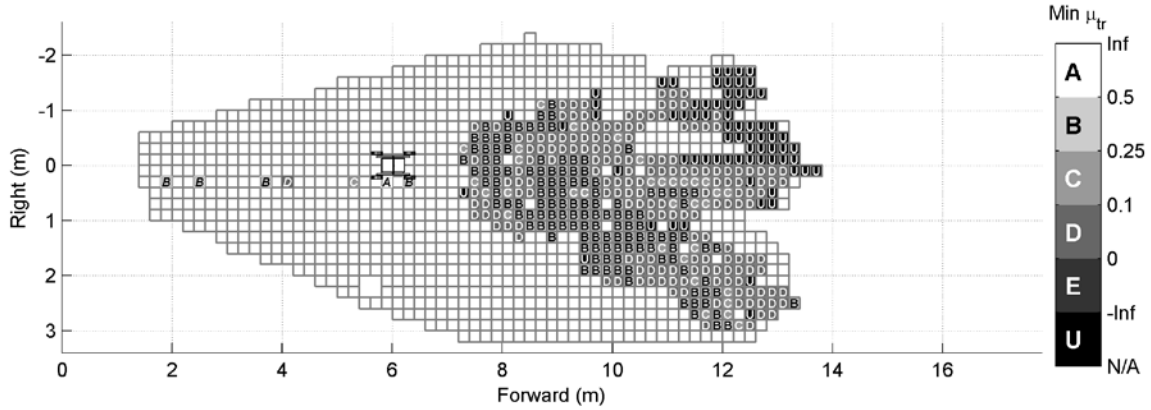


Figure 20: Terrain learning system results, at $t = 129.0$ sec, distance traveled = 3.4 m, (a) 3-D view, (b) plan view showing terrain classes

Figure 20 shows the rover's internal terrain map after 129 seconds, when the rover has traveled 3.4 meters. Figure 20(a) shows that the rover's knowledge of the terrain topography has increased, as illustrated by the reduced number of gaps in its internal representation of the terrain. Figure 20(b) shows that the rover has identified the minimum traction coefficient for three terrain patches with associated stereo data. These are illustrated in the figure by the three labeled terrain patches at 1.9m, 2.5m, and 3.7m. (Here, terrain patches that have been characterized through physical interaction are labeled with italicized letters.) Since these three patches all fall into class *B* (representing a lower bound of μ_{tr} between 0.25 and 0.5 from Table 4), all of the recognized (i.e. not novel) terrain in the distance is predicted to fall into that range. Terrain that is sufficiently different from the terrain the rover has driven over is still labeled "U," with unknown traversability properties.



(a)



(b)

Figure 21: Terrain learning system results, at $t = 226.5$ sec, distance traveled = 6.02 m, (a) 3-D view, (b) plan view showing classes

Finally, Figure 21 shows the rover's internal terrain map after 226.5 seconds. At this point the rover has traction data from terrain with a range of traction coefficients, and it has associated visual data with several locally identified classes. Thus, when the terrain is observed in the distance, a variety of traction coefficients are predicted. Some sections show high traction forces (labeled "B"), while others show lower traction forces (labeled "C" or "D"). Some terrain is still observed to be novel, suggesting that the terrain in the distance may have a significantly different appearance than the terrains previously traversed.

6 Conclusions—Self-Supervised Terrain Classification

This report has presented a self-supervised terrain classification framework to enable planetary rovers to predict mechanical properties of distant terrain. As components for this framework, both proprioceptive and exteroceptive terrain classifiers were developed and experimentally validated.

Two distinct proprioceptive terrain classification approaches were presented. A vibration-based classification approach was presented that is suitable for use in scenarios when labeled vibration training data is available. A novel traction-based classification approach was presented that is suitable for an unsupervised scenario, i.e. when the terrain classes are not known a priori. Additionally, an exteroceptive terrain sensing approach was developed that is suitable for unstructured environments.

The self-supervised classification framework was experimentally validated via implementation of a four wheeled rover operating in a beachfront environment. It was shown that the proposed approach exhibits performance that meets or exceeds performance of a system based on supervised classification.

Future work in this area would be useful to demonstrate the proposed algorithm in an environment where the traversability properties of the entire terrain map were independently measured. For implementation on a high-value system, such as a Mars rover, additional research would be necessary to identify how much training is necessary to yield a robust classifier. Additionally, the approach might be improved by use more robust rover localization methods than simple odometry, or by replacing the SVM classifier with another classifier designed for incremental training. The self-supervised learning framework may also be applied to other domains, such as high-speed UGVs or human-driven automobiles.

References—Self-Supervised Terrain Classification

- Angelova, A., Matthies, L., Helmick, D., & Perona, P. (2007). Learning and prediction of slip from visual information. *Journal of Field Robotics*, 24(3), 205-231. doi:10.1002/rob.20179
- Arvidson, R. E., Anderson, R. C., Bartlett, P., Bell, J. F., Blaney, D., Christensen, P. R., Chu, P., et al. (2004). Localization and Physical Properties Experiments Conducted by Spirit at Gusev Crater. *Science*, 305(5685), 821-824. doi:10.1126/science.1099922
- Azimi-Sadjadi, M. R., Ghaloum, S., & Zoughi, R. (1993). Terrain classification in SAR images using principal components analysis and neural networks. *Geoscience and Remote Sensing, IEEE Transactions on*, 31(2), 511-515.
- Bekker, M. G. (1969). *Introduction to Terrain-Vehicle Systems*. Ann Arbor: University of Michigan Press.
- Brooks, C. (2004). *Terrain Identification Methods for Planetary Exploration Rovers* (Thesis (S.M.)). Massachusetts Institute of Technology. Retrieved from <http://dspace.mit.edu/handle/1721.1/30303>
- Brooks, C. A. (2009, May 7). *Learning to Visually Predict Terrain Properties for Planetary Rovers* (Thesis (Ph.D.)). Massachusetts Institute of Technology.
- Brooks, C. A., & Iagnemma, K. (2005). Vibration-based terrain classification for planetary exploration rovers. *IEEE Transactions on Robotics*, 21(6), 1185-1191. doi:10.1109/TRO.2005.855994
- Brooks, C. A., & Iagnemma, K. (2009). Visual Detection of Novel Terrain via Two-Class Classification. *Proceedings of the 24th Annual ACM Symposium on Applied Computing 2009* (pp. 1145-1150). Honolulu, Hawaii: Association for Computing Machinery.

- Brooks, C. A., Iagnemma, K., & Dubowsky, S. (2006). Visual wheel sinkage measurement for planetary rover mobility characterization. *Autonomous Robots*, 21(1), 55-64. doi:10.1007/s10514-006-7230-9
- Chang, C.-C., & Lin, C.-J. (2005). *LIBSVM: a library for support vector machines*. Retrieved from <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- Chang, C.-C., & Lin, C.-J. (2008, October 29). LIBSVM: a Library for Support Vector Machines. Retrieved from <http://www.csie.ntu.edu.tw/~cjlin/reports/libsvm.pdf>
- Cowen, R. (2005). Opportunity Rolls out of Purgatory. *Science News*, 167(26), 413.
- Espinal, F., Huntsberger, T. L., Jawerth, B. D., & Kubota, T. (1998). Wavelet-based fractal signature analysis for automatic target recognition. *Optical Engineering*, 37(1), 166-174. doi:10.1117/1.601844
- Goldberg, S. B., Maimone, M. W., & Matthies, L. (2002). Stereo vision and rover navigation software for planetary exploration. *Proceedings of the 2002 IEEE Aerospace Conference* (Vol. 5, pp. 2025-2036). Big Sky, Montana.
- Grossman, L. (2010). Atom & cosmos: Spirit stuck, but in good spot. *Science News*, 177(5), 7.
- Halatci, I. (2006). *Vision-Based Terrain Classification and Classifier Fusion for Planetary Exploration Rovers* (Thesis (S.M.)). Massachusetts Institute of Technology. Retrieved from <http://dspace.mit.edu/handle/1721.1/38271>
- Halatci, I., Brooks, C. A., & Iagnemma, K. (2008). A Study of Visual and Tactile Terrain Classification and Classifier Fusion for Planetary Exploration Rovers. *Robotica*, 26(6), 767-779. doi:10.1017/S0263574708004360
- Iagnemma, K., & Buehler, M. (2006). Editorial for Journal of Field Robotics—Special Issue on the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9), 655-656.
- Iagnemma, K., Kang, S., Brooks, C., & Dubowsky, S. (2003). Multi-Sensor Terrain Estimation for Planetary Rovers. *Proceedings of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space, i-SAIRAS*. Nara, Japan.
- Iagnemma, K., Kang, S., Shibly, H., & Dubowsky, S. (2004). Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers. *IEEE Transactions on Robotics*, 20(5), 921-927. doi:10.1109/TRO.2004.829462
- Iagnemma, K., Shibly, H., & Dubowsky, S. (2002). On-line terrain parameter estimation for planetary rovers. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA '02)* (Vol. 3, pp. 3142-3147). Washington, DC. doi:10.1109/ROBOT.2002.1013710
- Jackel, L. D., Krotkov, E., Perschbacher, M., Pippine, J., & Sullivan, C. (2006). The DARPA LAGR program: Goals, challenges, methodology, and phase I results. *Journal of Field Robotics*, 23(11-12), 945-973.
- Kang, S. (2003). *Terrain Parameter Estimation and Traversability Assessment for Mobile Robots* (Thesis (S.M.)). Massachusetts Institute of Technology.
- Kim, D., Sun, J., Oh, S. M., Rehg, J. M., & Bobick, A. F. (2006). Traversability classification using unsupervised on-line visual learning for outdoor robot navigation. *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA '06)* (pp. 518-525). Orlando, FL.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-95)* (pp. 1137-1143). Montreal, Canada: Morgan Kaufmann. Retrieved from <http://robotics.stanford.edu/%7Eronnyk/accEst.pdf>

- Konolige, K. (2007). *SVS: Small Vision System*. Menlo Park, CA: SRI International. Retrieved from <http://www.videredesign.com/vision/svs.htm>
- Krebs, A., Pradalier, C., & Siegwart, R. (2010). Adaptive rover behavior based on online empirical evaluation: Rover-terrain interaction and near-to-far learning. *Journal of Field Robotics*, 27(2), 158-180.
- Matlab (Version 7.1) with Optimization Toolbox (Version 3.0.3)*. (2005). . Natick, MA: Mathworks.
- Mishkin, A., & Laubach, S. (2006). From Prime to Extended Mission: Evolution of the MER Tactical Uplink Process. *Proceedings of SpaceOps 2006 Conference* (pp. AIAA-2006-5689). Rome, Italy: American Institute of Aeronautics and Astronautics.
- Moore, H. J., Hutton, R. E., Scott, R. F., Spitzer, C. R., & Shorthill, R. W. (1977). Surface materials of the Viking landing sites. *Journal of Geophysical Research*, 82(28), 4497-4523. doi:10.1029/JS082i028p04497
- NASA/JPL. (2005, July 12). JPL.NASA.GOV: Pushing Away from Purgatory, Mars Exploration Rover Image Release. *NASA Jet Propulsion Laboratory*. Retrieved March 6, 2009, from <http://www.jpl.nasa.gov/missions/mer/images-print.cfm?id=1709>
- NASA/JPL. (2007). *MSL Landing Site Selection - User's Guide to Engineering Constraints. Version 4.5*. The Mars Science Laboratory Project. Jet Propulsion Laboratory, California Institute of Technology. Retrieved from http://webgis.wr.usgs.gov/msl/docs/MSL_Eng_User_Guide_v4.5.1.pdf
- Nilsson, N. J. (1982). *Principles of Artificial Intelligence*. New York, N.Y.: Springer-Verlag.
- Olsen, R., Garner, J., & Van Dyke, E. (2002). Terrain classification in urban wetlands with high-spatial-resolution multispectral imagery. *Proceedings of SPIE Conference on Sensors, Systems, and Next-Generation Satellites VI* (Vol. 4881, pp. 686-691). Agia Pelagia, Crete, Greece: SPIE. doi:10.1117/12.465161
- Rasmussen, C. (2002). Combining laser range, color, and texture cues for autonomous road following. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA '02)* (Vol. 4, pp. 4320-4325). Washington, DC. doi:10.1109/ROBOT.2002.1014439
- Reina, G., Ojeda, L., Milella, A., & Borenstein, J. (2006). Wheel slippage and sinkage detection for planetary rovers. *IEEE/ASME Transactions on Mechatronics*, 11(2), 185-195. doi:10.1109/TMECH.2006.871095
- Rover Team. (1997). Characterization of the Martian Surface Deposits by the Mars Pathfinder Rover, Sojourner. *Science*, 278(5344), 1765-1768. doi:10.1126/science.278.5344.1765
- Sofman, B., Lin, E., Bagnell, J. A., Cole, J., Vandapel, N., & Stentz, A. (2006). Improving robot navigation through self-supervised online learning. *Journal of Field Robotics*, 23(11-12), 1059-1075. doi:10.1002/rob.20169
- Stentz, A. (1994). *The D* Algorithm for Real-Time Planning of Optimal Traverses* (Technical Report No. CMU-RI-TR-94-37) (p. 34). Pittsburgh, PA: The Robotics Institute, Carnegie Mellon University. Retrieved from http://www.ri.cmu.edu/pubs/pub_356.html
- Strang, G. (1993). Wavelet Transforms Versus Fourier Transforms. *Bulletin of the American Mathematical Society*, 28(2), 288-305. doi:10.1090/S0273-0979-1993-00390-2
- Talukder, A., Manduchi, R., Castano, R., Owens, K., Matthies, L., Castano, A., & Hogg, R. (2002). Autonomous terrain characterisation and modelling for dynamic control of unmanned vehicles. *Proceedings of the 2002 IEEE/RSJ International Conference on*

- Intelligent Robots and System* (Vol. 1, pp. 708-713). Lausanne, Switzerland. doi:10.1109/IRDS.2002.1041474
- Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., et al. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9), 661-692. doi:10.1002/rob.20147
- Videre Design. (2001). STH-DCAM Stereo Head User's Manual, Rev. 2. Retrieved from <http://www.videredesign.com/docs/sthdcam.pdf>
- Welch, P. D. (1967). The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2), 70-73.
- Weszka, J. S., Dyer, C. R., & Rosenfeld, A. (1976). A Comparative Study of Texture Measures for Terrain Classification. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-6(4), 269-285. doi:10.1109/TSMC.1976.5408777
- Wilcox, B. H. (1994). Non-geometric hazard detection for a Mars microrover. *Proceedings of the Conference on Intelligent Robotics in Field, Factory, Service, and Space (CIRFFSS '94)*, NASA Conference Publication (Vol. 2, p. 675). Houston, TX, USA: Publ by NASA, Washington, DC, USA.
- Wong, J.-Y. (2001). *Theory of Ground Vehicles* (3rd ed.). New York: John Wiley.
- Wong, J.-Y., & Reece, A. R. (1967). Prediction of rigid wheel performance based on the analysis of soil-wheel stresses part I. Performance of driven rigid wheels. *Journal of Terramechanics*, 4(1), 81-98. doi:10.1016/0022-4898(67)90105-X

7 Introduction—Self-Supervised Road Detection

Other work under this grant has focused on novel applications of the self-supervised terrain learning approach being developed under this research grant. One application of particular interest is urban/semi-urban driving on road networks. For many UGV applications (including convoy applications), it would be desirable to automatically identify the drivable road surface location, and analyze and adapt to changing road conditions. We have developed a method for accomplishing this, which was inspired by the work previously done under this grant for off-road navigation.

8 Road Detection Based On Online Learning And Evaluation

Road detection is an important requirement for the successful development and employment of UGVs. In the past decades, research on vision-based road detection has been an active topic and various methods have been proposed to solve this problem [1]-[5]. In principle, vision-based road detection algorithms can be categorized into three main classes: feature-based techniques [1][2], model-based techniques [3] and region-based techniques [4]-[6]. Generally speaking, feature based technique are more accurate than any others, but they require the detected road to have well-painted markings. Model-based techniques can be more robust than feature-based techniques. However, most model-based approaches employ some strict geometrical assumptions. Most effective approaches to region-based techniques also can be seen as a machine learning problem. Those approaches allow computers to change behavior based on comparison of the candidate road to a training set, and are capable of being robust to noise.

For UGV operation in real-world scenarios, the environment is continually changing. Thus, a major difficulty with machine learning-based approaches is how to train the algorithm online, to be capable of adapting to the new environment. Wang [4] trained an SVM classifier in initialization and used a voting method to determine the correct class online. This required human supervised learning in every frame. Foedisch [6] selected training data in each frame by dynamic windows which adaptively adjusted their positions based on the result of the last frame. This algorithm is not adaptive in some situations as shown in Fig.1. In Fig.1, the sky is labeled as road because the pixels in the sky region haven't been explicitly included during training of the classifier. The drawback of dynamic window-based algorithm is that the windows simply adjust their positions according to the region, but not the properties of the training data. If the windows fail to cover some training set which determines the hyperplane in the real feature space, the result would be corrupted.

Fig.2 shows two examples of training sets and test sets in a feature space. There, pink points and dark blue points in (2) and (4) are positive and negative training sets, respectively. Red points and light blue points in (1), (3) and (5) are positive and negative test sets, respectively. Black points are misclassified data.

In this report, we describe a novel road detection algorithm that is capable of not only

performing online evaluation, but also automatically selecting the training set which has more contribution in determining the correct classification hyperplane.

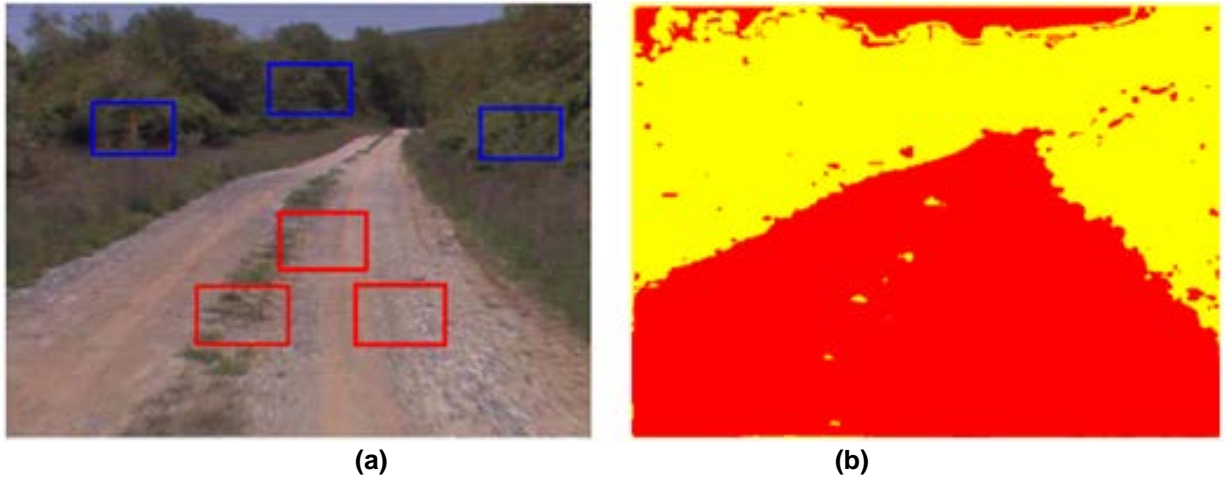


Fig.1 Road detection based on Dynamic Windows Method: (a): Selecting the pixels in the windows as the training data to train the classifier. (b): Classification result by the trained classifier

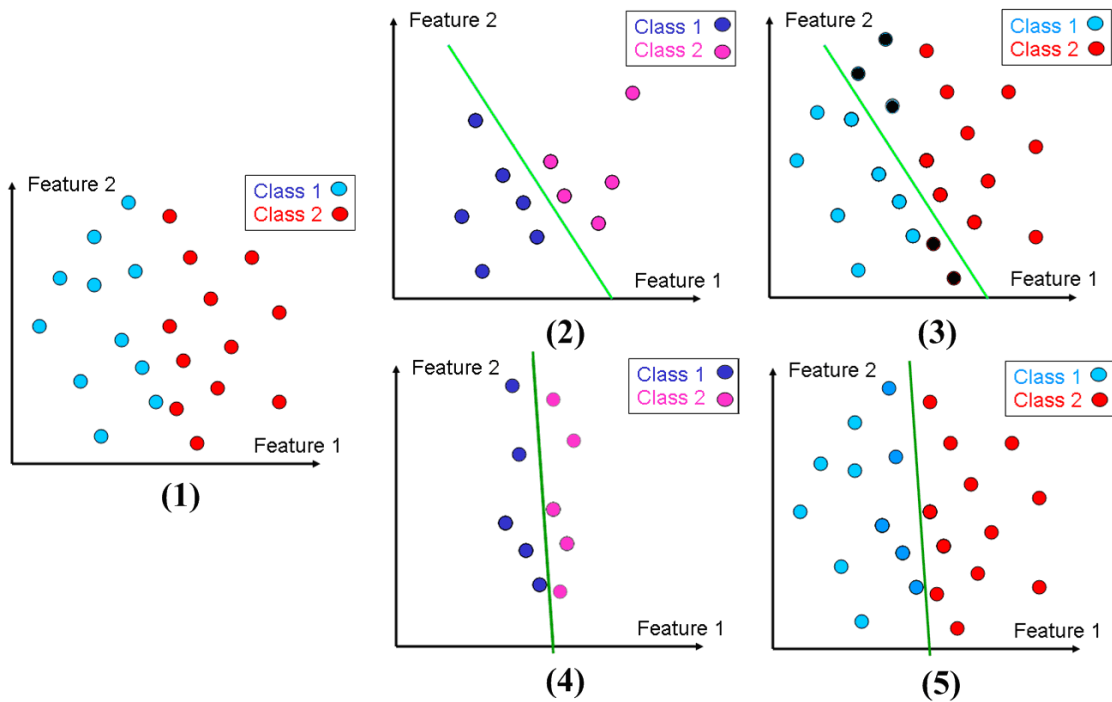


Fig.2 Example Feature Space and Hyperplane

8.1 Algorithm Outline

The proposed algorithm is composed of five components. In the first, feature extraction component, a feature vector is extracted from each pixel of the input image. Second, a dynamic training database (DTD) is filled with training sets labeled by a human supervisor during initialization, and is updated by the new training set online. Third, classifier parameters are computed to estimate the parameters in an SVM classifier. Fourth, the SVM classifier classifies online image into road/non-road classes. The last component contains two stages: Morphological Operation and Online Learning Operation. The former implements connected region growing and hole filling on the classification result to determine the road region. The latter compares morphological result and classification result to evaluate the quality of current classifier, then select new training set from that comparison and update the DTD.

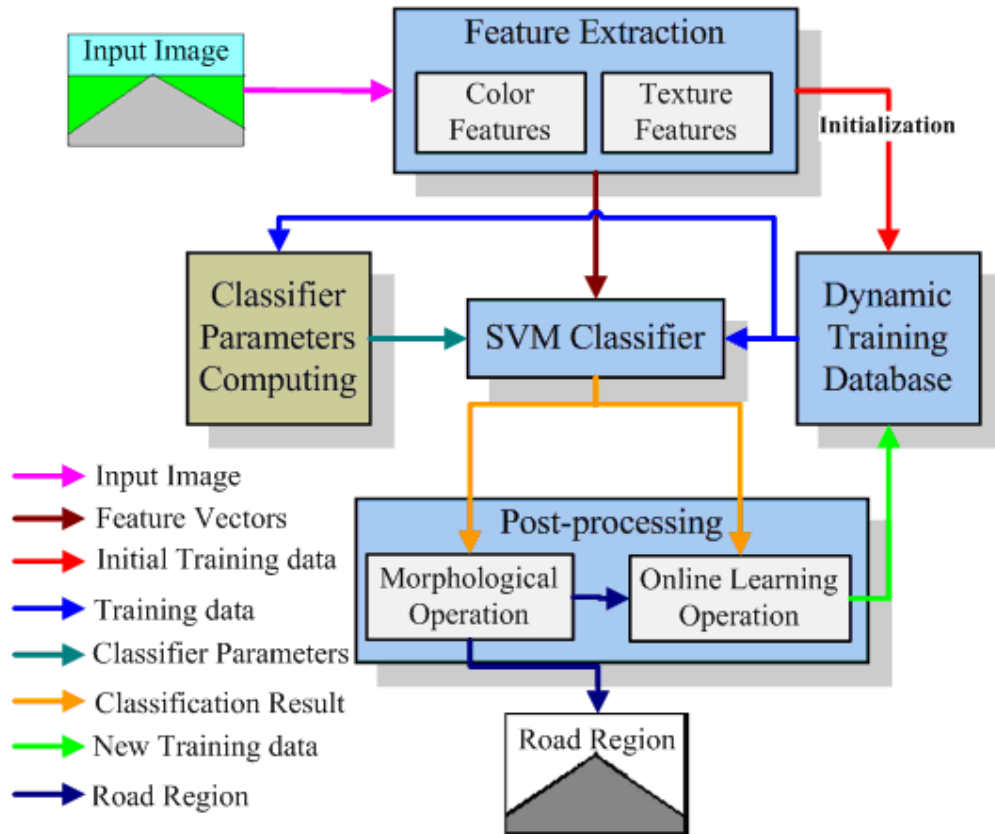


Fig.3 Algorithm flowchart

We will introduce feature extraction, initial DTD, classifier parameters computing, SVM classifier and morphological operation in Section 2.2. Those can be seen as the basic parts of our algorithm. Then, in Section 3.0, we will discuss Online Learning Operation and Online DTD, which are the advanced parts of our algorithm. The advanced parts make our algorithm more robust and adaptive to environment changing. In Section 4.0, we will focus on the experimental results.

8.2 Basic Parts of Road Detection Algorithm

The basic parts of road detection algorithm are as follows. First, the feature vector in each pixel of image is extracted. Then, two small windows are labeled by a human supervisor to sample the training set to initialize the DTD. Then, the training data is used for classifier training. Last, the trained classifier is implemented to classify the road/non-road classes and the morphological operation is taken to smooth the road region.

8.2.1 Feature Extraction

The visual features used in our algorithm are color features and texture features. Color data is directly available from the camera as RGB intensities. However, the illumination intensity affects all three values in a raw RGB representation which can lead to poor classification results. To reduce that effect, a HSV color representation is used. Texture is a measure of the local spatial variation in the intensity of an image. In this report, the first five Haralick statistical features [7] are exploited. Those three color features and five texture features are combined to form an eight-element feature vector as following:

$$F_{i,j} = [f_{t_1(i,j)}, \dots, f_{t_5(i,j)}, f_{c_1(i,j)}, \dots, f_{c_3(i,j)}] \quad i=1, \dots, H \quad j=1, \dots, W \quad (1)$$

where $f_{t_n(i,j)}$ is the n^{th} Haralick statistical feature at the point (i, j) , $f_{c_n(i,j)}$ is the n^{th} color feature at the point (i, j) in HSV color space, the H and W are the height and width of image.

8.2.2 Initial Dynamic training database

There are two stages to build the dynamic training database: an initial stage and an online learning stage. The latter will be discussed in next Section. In the initial stage, the training data is labeled by a human supervisor. Two windows are placed on the image by the supervisor to select the training data as shown in Fig. 3.

The feature vectors of pixels in these windows are outputted into the DTD. To reduce the computation, the size of DTD we used in our algorithm is limited as 1000. If any window contains more than the maximum size, a random function is used to select samples.

8.2.3 Classifier Parameter Computing

Obviously, the relation between road/non-road classes and their feature space is nonlinear. In that case, we cannot find a linear hyperplane to separate two classes in the feature space. A suggestion to use a Gaussian radial basis function (RBF) is recommended by [8]. Here a RBF kernel is used as the SVM kernel function. There are two classifier parameters associated with this kernel: C and γ . It is not known beforehand which C and γ are the best for the road detection in a new environment. The goal of Classifier Parameters Computing is to identify good (C, γ) so that the classifier can accurately predict unknown data (i.e., testing data). Therefore, in

Classifier Parameter Computing, we use cross-validation over the training data from DTD and grid-search on those two classifier parameters (see [8] for the details).



Fig.3 Dynamic Training database initialization Red windows are for positive training set selection. Blue windows are for negative training set selection.

The outdoor environment is changing continuously while the vehicle is moving. However, because the process of classifier parameter computing is time and computation consuming, it is not suitable to compute the parameters in every frame. Therefore, given the assumption that the environment does not change drastically in a few consecutive frames, we take the classifier parameter computing as a parallel process with all other components. From the experiments, the values of parameters are updated in every 8-12 frames.

8.2.4 SVM Classifier

There are two stages in the SVM Classifier: road detection classifier training and road detection classifier classification.

8.2.5 Road detection classifier training:

Fig. 4 gives an outline of the road detection classifier training stage. As mentioned above, we use the SVM with the RBF kernel as the road detection classifier. Given the classifier parameters by classifier parameter computing and the training data by DTD, the SVM classifier determines the separating hyperplane with largest margin in the high-dimensional feature space (See [9] for the

details of SVM).

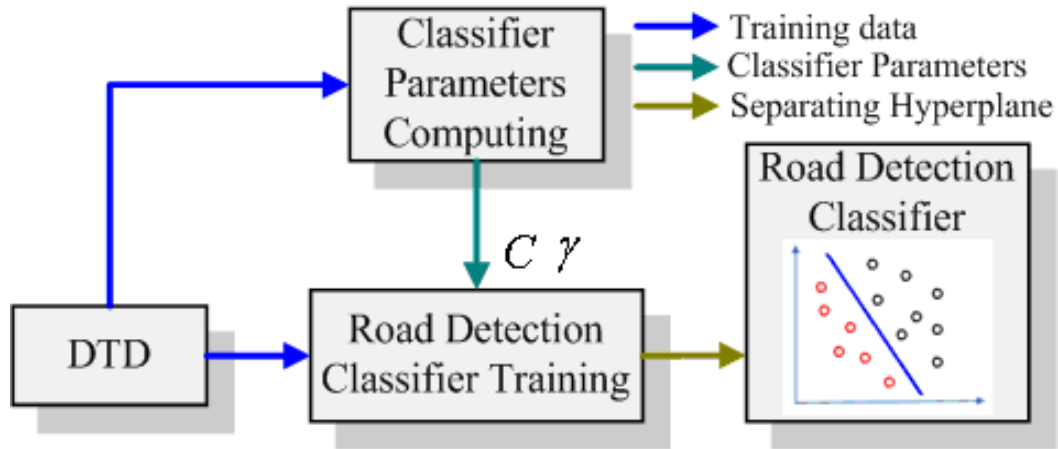


Fig.4 Road Detection Classifier Training Outline

8.2.6 Road detection classifier classification

Fig.5 shows an outline of the classification stage. We extract the feature vector from each pixel of the input image. Then, each vector is classified by the trained classifier.



Fig.5 Road detection classifier classification Outline

Each pixel is classified as belonging to either the road class or non-road class. Fig 6-a shows the sampling windows placed by human supervisor. Then the pixels in the sampling windows are used as training data to train the road detection classifier. Fig. 6-b gives the classification results.

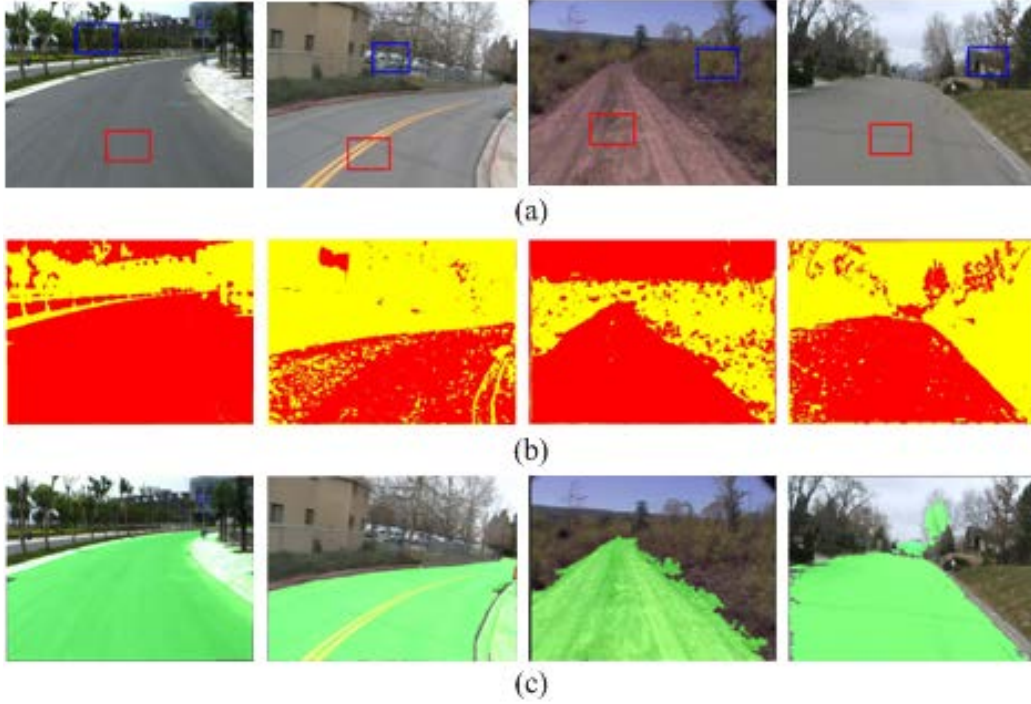


Fig.6 Flow of basic road detection algorithm. (a) The original image and sampling windows. (b) The classification results (Red is road class. Yellow is non-road class). (c) The results of morphological operation. (The regions labeled as green are road regions).

8.2.7 Morphological operation

Morphological operations [10] are commonly used to understand the structure or form of an image. Morphological operations play a key role in applications such as machine vision and automatic object detection.

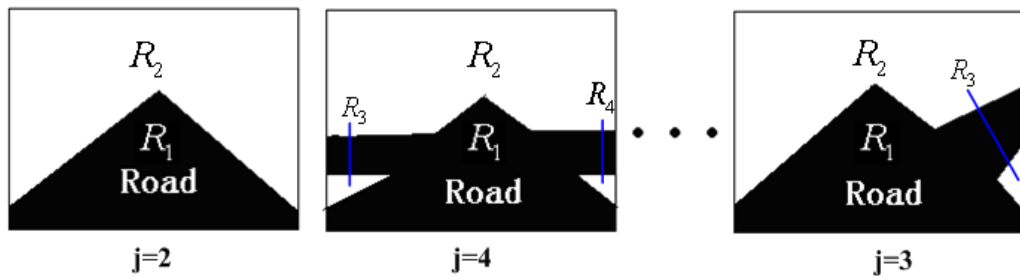


Fig.7 Simply connected road model (R_1 is simply connected road region. $R_n (n > 1)$ is non-road region.)

Here, the main morphological operation is a flood-fill operation based on the assumption that the road region is simply connected (as shown in Fig 7). The morphological operation is implemented to determine the largest connected road region and erode the holes in that connected road region. Then that largest connected road region is labeled as the road region and others are labeled as non-road regions. The process of morphological operation is showed in

Fig.8. Fig.6-c shows the results after morphological operation.

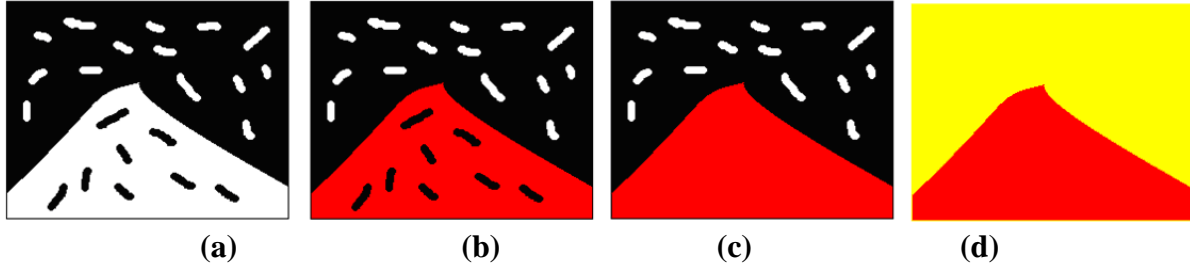


Fig.8 Morphological Operation (a): Classification result (white: represents road; black: represents non-road). (b): Largest connected road region (Red). (c): Erosion operation. (c): Morphological operation result (red is road region and yellow is non-road region)

8.3 Advanced Parts Of Road Detection Algorithm

The advanced parts of the road detection algorithm can be seen as the crucial parts in the framework of self-supervised online learning. From Fig.6-b, it can be seen that the basic road detection algorithm misclassified many points on the image. Although morphological operation can help get rid of most misclassified points (see Fig.6-c), in the long run the misclassified points would bring a potentially dangerous situation when the misclassified points connected with the road region. (The curbs were connected with road regions in the first two images of Fig.6-c. The tree was misclassified as road in the fourth image of Fig.6-c.).

A key reason for misclassification is that the training data didn't properly represent the test feature vector space. That is also the main drawback of window-based learning algorithm which the window just can find the local feature in the entire image. The task of self-supervised online learning is to automatically find interesting training samples which haven't be learned and but are important in road/non-road classes determination.

In this section, we will introduce two crucial components to solve the automatic learning process in our algorithm: Online Learning Operation and DTD in online learning stage.

i. Online Learning Operation

1. Evaluation function

Before executing the online learning operation, an evaluation function is implemented to evaluate the performance of current classification and determine whether to activate the Online Learning Operation and train the road detection classifier in the next frame. This evaluation functions as shown in the following formulas are based on the assumption we mentioned in previous sections that the road region is simply connected.

$$E_{AFP} = \frac{\sum_{r=1}^H \sum_{c=1}^W V_1(r, c)}{\sum_{r=1}^H \sum_{c=1}^W R_1^M(r, c)} \quad (2)$$

$$E_{AFN} = \frac{\sum_{j=2}^N \sum_{r=1}^H \sum_{c=1}^W V_j(r, c)}{\sum_{j=2}^N \sum_{r=1}^H \sum_{c=1}^W R_j^M(r, c)} \quad (3)$$

$$E_{AF} = \frac{\sum_{j=1}^N \sum_{r=1}^H \sum_{c=1}^W V_j(r, c)}{\sum_{j=1}^N \sum_{r=1}^H \sum_{c=1}^W R_j^M(r, c)} \quad (4)$$

$$V_j(r, c) = \begin{cases} 1, & \text{if } (R_j^C(r, c) \neq R_j^M(r, c)) \\ 0, & \text{if } (R_j^C(r, c) = R_j^M(r, c)) \end{cases} \quad r = 1, \dots, H; c = 1, \dots, W; j = 1, \dots, N \quad (5)$$

where AFP, AFN and AF refer to Assumption-based False Positive, Assumption-based False Negative and Assumption-based classification False, the N is the number of regions which are defined in Fig.7, the r and c are the row and column, the $R_j^C(r, c)$ is the value of classification result at (r, c) , the $R_j^M(r, c)$ is the morphological operation result at (r, c) , the j is the number of region., $V_j(r, c)$ indicates whether $R_j^C(r, c)$ and $R_j^M(r, c)$ belong to the same class. Given the values of E_{AFP} , E_{AFN} and E_{AF} , we have three thresholds of $T_{E_{AFP}}$, $T_{E_{AFN}}$ and $T_{E_{AF}}$ to tune. If the value of evaluation is larger than its threshold, the retraining process is implemented.

2. Online Learning

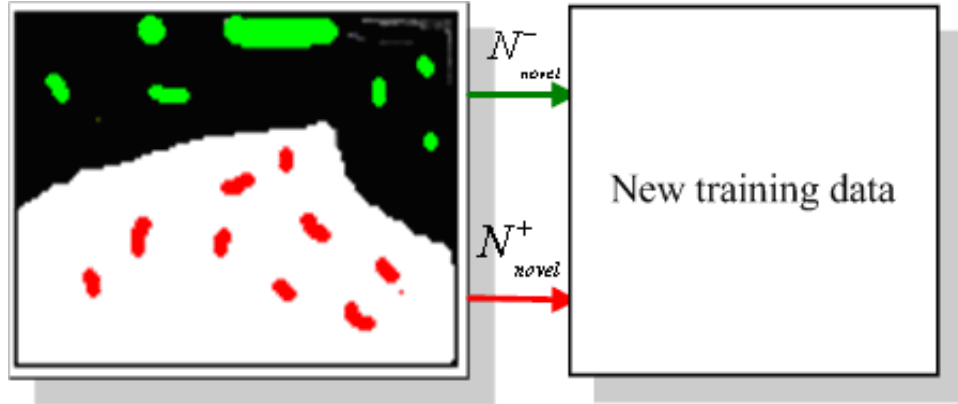


Fig.9 Interesting points. Green points are in the non-road region and misclassified as road class. Red points are in the road region and misclassified as non-road class.

The process of online learning in our algorithm is to acquire feature vectors at interesting, novel points. Given that assumption that the road region is simply connected, the points classified as road lying in the regions of non-road can be seen as interesting points and are labeled as negative samples (non-road samples). We label those points as new training data (shown in Fig.9). In practice, one doesn't know exactly where the real boundary of the road region is located. We can detect the edge of road region in the morphological operation result. In order to reduce the possibility of mislabeled training data near the road boundary, a threshold M is set as the width of margin near boundary which segments the road and non-road region in a morphological operation result as shown in Fig.10. In our experiment, we set M as 40 pixels width.



Fig.10 Road region boundary in Morphological Operation Result

3. DTD in online learning stage

In order to reduce the computation of training the SVM classifier, the DTD is limited to contain 1000 training positive and 1000 negative samples labeled by the human supervisor in the initial stage. Usually, the number of new training data in each class is more than 1000. So we randomly choose T new training samples in each class and also abandon T old training samples in each class in DTD. T is a threshold to determine the learning speed. Too large a value of T will lead to over training on new misclassified data, while too small value of T will give our algorithm low adaptability and robustness to changing environment conditions. From the viewpoints of our many experimental results, it is best to set T as 1/10 of the size of DTD (T is 100 in this report).

8.4 Results

Road detection by the proposed algorithm worked well in a variety of test conditions. First, some results are shown to demonstrate how the adaptive online learning process acquires training data. Then, a comparison between our algorithm and a dynamic window approach is shown. At last, the results in sequences which were taken in different conditions are shown. The system's performance in the following results is compared with manually annotated frames to measure the accuracy.

i. Comparison between offline learning and online learning

Results from four different conditions are shown in Fig.11. In each experiment, two small sampling windows are selected on the image to initialize the DTD. This can be seen as offline learning. Then, our algorithm restudies the poor classification result, and retrain the classifier and reclassifies the road image.

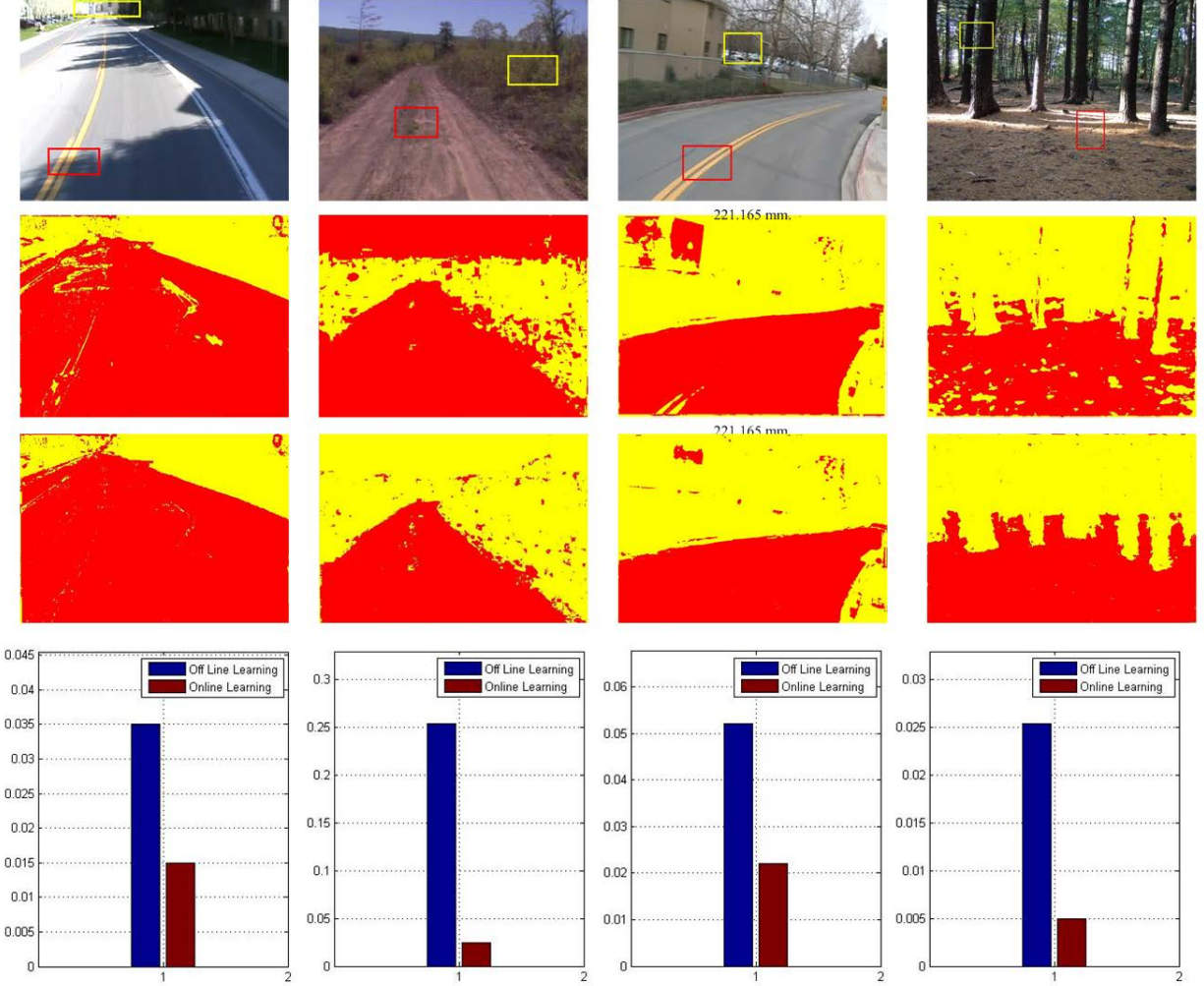


Fig.11 Comparison results between Offline learning and online learning

The first row shows original images. The second row shows the classification results of offline learning. The third row shows the classification results of online learning. The fourth row shows the classification error rates.

The first column shows the results of different learning processes in the shadowed road situation. From the offline learning results, some road marks and tree shadows are misclassified as non-road. After the online learning process, the number of misclassified points is reduced.

The second column shows the results in the unstructured road situation. Almost all the points in the sky are misclassified as road in the result of offline learning. With the process of online learning, the accuracy is significantly improved.

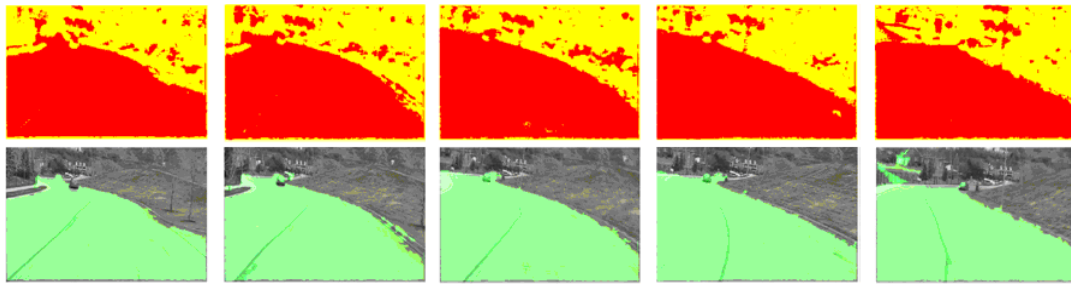
From the third column we can see that the wall of the left building is misclassified as road due to a lack of samples in the offline sampling windows. After our self supervised online learning

process, the computer automatically learns the new samples and generates a more accurate result.

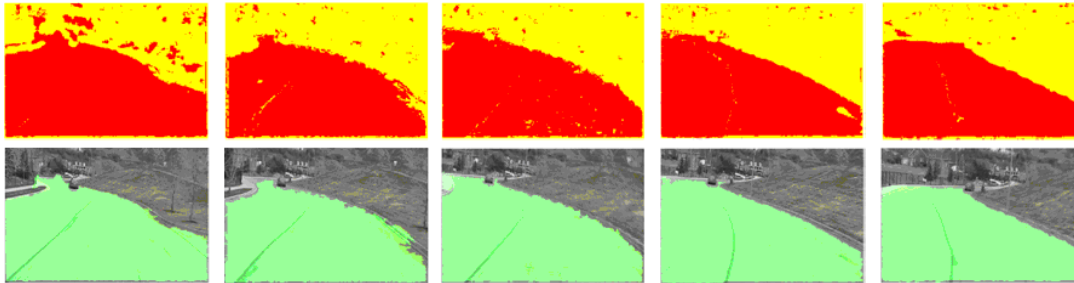
It is more complicated for road detection in the forest. However, from the results in the fourth column we could see that the points are well classified and the classification error becomes smaller after our online learning process.

ii. *Comparison with dynamic windows approach in consecutive frames*

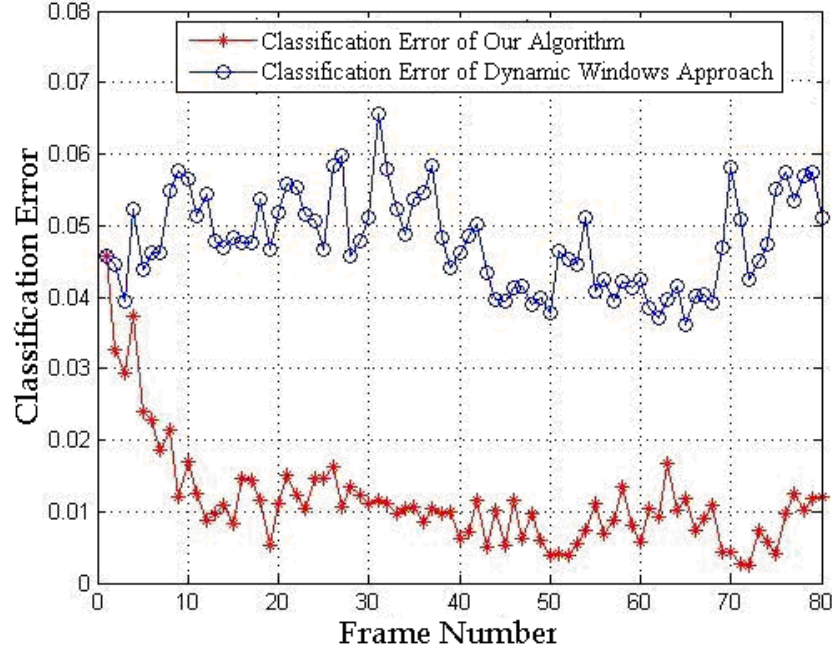
We compare our algorithm with a dynamic window approach [6] in consecutive video frames. Some frames of classification results and morphological operation results are shown in Fig.12-a, Fig.12-b. The classification result in each frame is compared with hand labeled ground truth to acquire the classification error (See Fig.12-c). From this brief results, we can see that our algorithm exhibits slightly better accuracy than a dynamic window approach.



a) Result of Dynamic Window Approach



b) Result of Proposed Approach



c) Classification Error

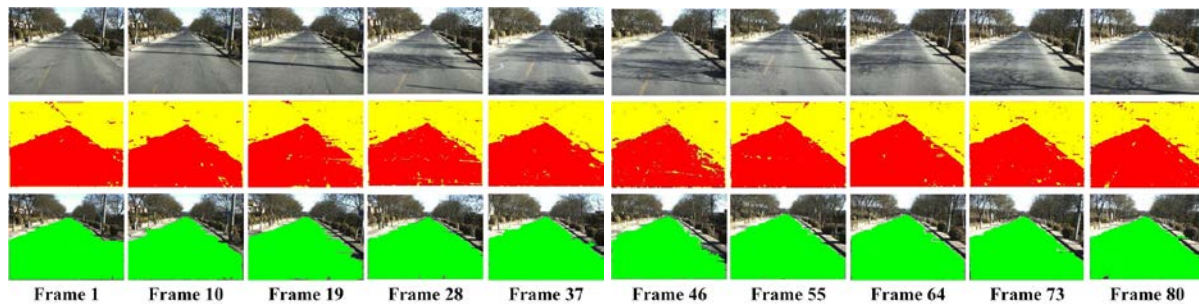
Fig.12 Comparison of Two Road Detection Approaches

iii. *Results of our algorithm in consecutive frames*

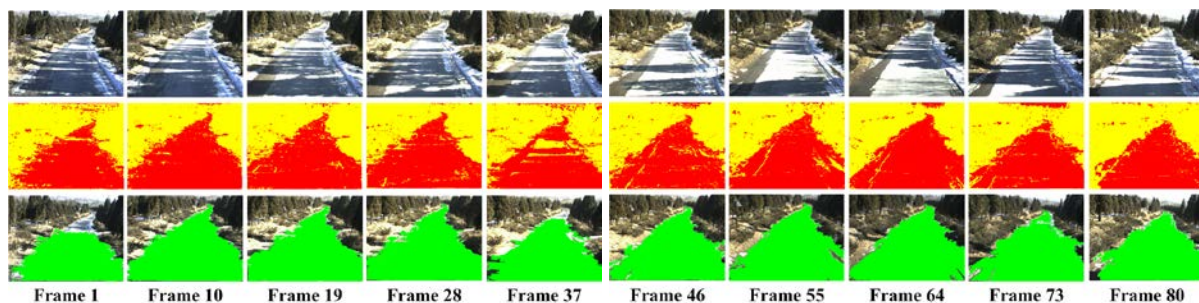
Fig.13 shows the results of our algorithm over in consecutive frames, in different situations. From the 7th and 61st frames in Fig.13-c, and the 37th and 69th in Fig.13-d we can see the classification error rates become sharply larger due to the changing environment, however our algorithm reduces the error by online learning in subsequent frames.

8.5 Conclusions—Self-Supervised Road Detection

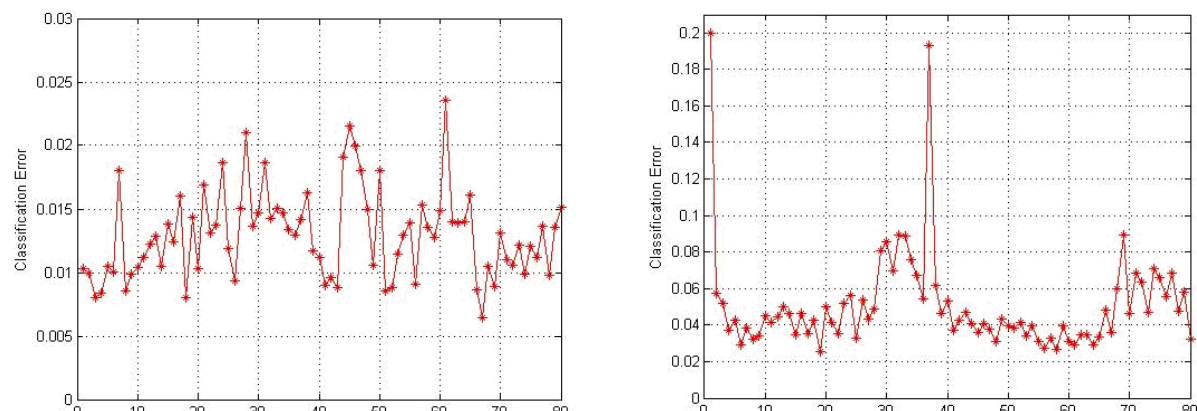
We introduced an algorithm for adaptive, robust online learning for road identification. In the future, we will combine a LIDAR sensor with the online learning process to instruct the training data acquisition. In order to speed up the learning convergence process, we also will find a method to abandon the old training data which are less important in classifier hyperplane determination in DTD (instead of randomly discarding).



(a)



(b)



9 References—Self-Supervised Road Detection

- [1] Tsai, L.W., Hsieh, J.W., Chuang, C.H., Fan, K.C.: Lane detection using directional random walks. In: IEEE Intelligent Vehicles Symposium. 2008.
- [2] Li, Q., Zheng, N., Cheng, H.: A prototype autonomous vehicle. Proceedings of the IEEE Machine Learning and Cybernetics, 2009 International Conference on, 67, pp. 786-804, 1979.
- [3] Y. Wang, E. Teoh, and D. Shen, Lane detection and tracking using B-snake, Image Vision Computing. vol. 22, no. 4, pp. 269–280, Apr. 2004.
- [4] Jian Wang; Zhong Ji; Yu-Ting Su . Unstructured road detection using hybrid features. Machine Learning and Cybernetics, 2009 International Conference on Volume 1, 12-15 July 2009 Page(s):482 – 486.
- [5] Sha Yun; Zhang Guo-ying; Yang Yong, A Road Detection Algorithm by Boosting Using Feature Combination. Intelligent Vehicles Symposium, 2007.
- [6] Foedisch, M; Takeuchi, A, Adaptive road detection through continuous environment learning. Applied Imagery Pattern Recognition Workshop, 2004.
- [7] Haralick, R.M. Statistical and Structural Approaches to Texture. Proceedings of the IEEE Machine Learning and Cybernetics, 2009 International Conference on, 67, pp. 786-804, 1979.
- [8] Chang, C., & Lin, C. (2008, October 29). LIBSVM: a Library for Support Vector Machines. Retrieved November 20, 2008, from <http://www.csie.ntu.edu.tw/~cjlin/reports/libsvm.pdf>.
- [9] Christopher J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery 2:121–167, 1998.
- [10] Soille, P., Morphological Image Analysis: Principles and Applications. Springer-Verlag, 1999, pp. 173-174.

10 Introduction—Experimental Analysis of Wheel-Terrain Interaction

Robotic vehicles are frequently deployed in unwelcoming, hazardous environments. From military robots to planetary rovers, vehicle mobility is a key aspect of mission success. Several models for traction modeling of tracked and wheeled vehicles have been developed in the past decades; however, a comprehensive understanding of soil behavior under running gear is still missing to date. The work of Bekker and Wong, which began in the 1950's, has laid the foundation for modern terramechanics. The application of classical results from plasticity theory, combined with semi-empirical formulations, has provided satisfactory solutions to the problem of mobility modeling for large, heavy vehicles. However, the expanded use of lightweight vehicles (especially man-portable robotic vehicles) has called for a new effort in modeling vehicle-terrain interaction problems. In fact, some researchers have suggested that classical models are of questionable utility when applied to vehicles one order (or more) of magnitude smaller than tanks, Humvees, large trucks, and the like [1].

This report will describe novel experimental methods aimed at understanding the fundamental phenomena governing the motion of lightweight vehicles on dry, granular soils. A single-wheel test rig is used to empirically investigate wheel motion under controlled wheel slip and loading conditions on a sandy, dry soil (Figure 1). Test conditions can be designed to replicate typical field scenarios for lightweight robots, while key operational parameters such as drawbar force, torque, and sinkage are measured. This test rig enables imposition of velocities, or application of loads, to interchangeable running gears within a confined soil bin of dimensions 1.5 m long, 0.7 m wide, and 0.4 m deep. This allows testing of small-scale wheels, tracks, and cone or plate penetrators.

The soil under investigation has been fully characterized with a series of direct shear tests (ASTM D3080) and penetration tests. Direct shear tests were performed to estimate soil shearing parameters such as cohesion, angle of internal friction, and shear modulus. Penetration tests, although not standard tests, were performed to evaluate 'Bekker' parameters, necessary for characterization of pressure-sinkage behavior of the soil under the methodology described by Wong [2].

The aforementioned experiments represent a typical experimental approach to macro-scale characterization of wheel-soil interaction. However, the application of classical terramechanics model to lightweight vehicles may potentially show discrepancy between experiments and predictions, warranting the development of new methods to probe the fundamental mechanics of a small robot's interaction with soil.

To this end, two additional experimental methodologies have been developed. The first relies on high-speed imaging of the wheel-soil interface and the use of particle image velocimetry (PIV) to measure micro-scale terrain displacement (Figure 1). This methodology, although confined to plane strain cases, allows measurement of soil flow velocities, and observation of the formation of shear bands beneath the wheel/track. Though this method does not explicitly permit calculation of the velocities of individual soil particles, it does allow estimation of a regularly-spaced velocity field in the soil. While such visualization techniques have been widely employed in the field of experimental fluid mechanics, their application to the study of soils is a relatively new development [3, 4].

The second experimental methodology is intended to complement the PIV-based soil kinematics analysis. It employs a custom force sensor array located at the wheel-terrain interface. The force sensors are strain gauge-based flexural elements with interchangeable

interface surfaces that are designed for integration with wheels or other running gear. The sensors allow explicit measurement of normal and shear forces (and, therefore, estimation of normal and shear stresses) at numerous discrete points along the wheel-soil interface. When coupled with PIV-derived kinematic data, this allows for a richer characterization of soil loading and failure regimes than would be possible with either kinematic or pressure information alone. In particular, this experimental methodology allows joint visualization of the soil displacement in the bulk soil medium, and measurement of shear and normal stress at points along the interface. This could lead to development and validation of novel constitutive relations describing soil behavior under loading imposed by running gear.

Experiments have shown that soil failure, at certain slip levels, is qualitatively different under cases of low vertical load (which is typical for lightweight robots) compared to cases of high vertical load (typical for large ground vehicles). Also, soil flow patterns have been observed to exhibit periodic failure phenomena, giving rise to interesting features such as surface ripple formation. These results, obtained through PIV analysis, provide deeper understanding of the mechanics of traction generation. Experimental measurements gathered by these test methodologies are compared against the results from well-established semi-empirical models, to understand limitations of these models and propose modifications and improvements.

11 Single Wheel Testbed Description

The Robotic Mobility Group at MIT has designed and fabricated a multipurpose terramechanics rig based on the standard design described by Iagnemma [5]. The testbed is pictured in Figure 1 and is composed of a Lexan soil bin surrounded by an aluminum frame where all the moving parts, actuators and sensors are attached. A carriage slides on two low-friction rails to allow longitudinal translation while the wheel or track, attached to the carriage, is able to rotate at a desired angular velocity. The wheel mount is also able to translate in the vertical direction. This typical setup allows control of slip and vertical load by modifying the translational velocity of the carriage, angular velocity of the wheel, and applied load. Horizontal carriage displacement is controlled through a toothed belt, actuated by a 90W Maxon DC motor while the wheel is directly driven by another Maxon DC motor. The motors are controlled thorough two identical

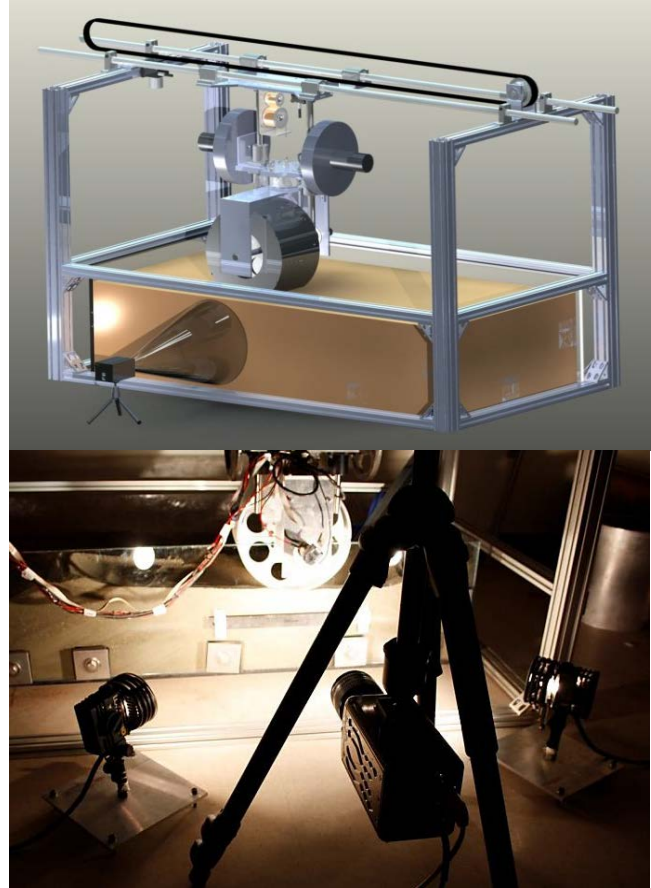


Figure 1: CAD drawing of the terramechanics testbed showing the imager for PIV experiments (top). Actual PIV setup with the high speed camera and two flood lights (bottom).

Maxon ADS 50/10 4-Q-Dc servoamplifiers. The carriage horizontal displacement is monitored with a Micro Epsilon WPS-1250-MK46 draw wire encoder while wheel vertical displacement (i.e., sinkage) is measured with a Turck A50 draw wire encoder. A 6-axis force torque ATI Omega 85 transducer is mounted between the wheel mount and the carriage in order to measure vertical load and traction generated by the wheel. Finally, a flange-to-flange reaction torque sensor from Futek (TFF500) is used to measure driving torque applied to the wheel. Control and measurement signals are handled by a NI PCIe-6363 card through Labview software.

The rig is capable of approximately 1 meter of horizontal displacement at a maximum velocity of approximately 120 mm/s with a maximal wheel angular velocity of approximately 40 deg/s. The bin width is 0.6 meters while the soil depth is 0.16 meters. Considering the wheel sizes and vertical loads under study, these physical dimensions are sufficient for eliminating boundary effects. Moreover, the same testbed, with some adaptations, can be used to perform soil penetration tests and analyze different running gears (e.g., both wheels and tracks).

For the experiments described in this report, the Mojave Martian Simulant (MMS) was employed as a test medium [6]. MMS is a mixture of finely crushed and sorted granular basalt intended to mimic, both at chemical and mechanical level, Mars soil characteristics. MMS particle size distribution spans from micron level to mm level with 80% of particles above the 10 micron threshold.

12 Granular Soil Particle Image Velocimetry

Particle image velocimetry (PIV) describes an experimental method, based on image cross-correlation techniques, used for the determination of flow velocity fields. The use of PIV for the calculation of fluid velocities initially emerged in the 1980's [7, 8]. Since then, PIV has played an important role in many fluid mechanics investigations [9]. Two of the main advantages of PIV over other methods for the measurement of velocity (e.g. hot-wire-velocimetry, Pitot tubes etc.) are that it is non-intrusive, and allows for relatively high resolution measurements over an extended spatial domain.

During fluid-based PIV analysis, the fluid is typically seeded with marker particles that refract, absorb, or scatter light, have a high contrast with the fluid, and do not interrupt the fluid flow. Imaging is performed at high speed over an area of the flow illuminated by a light source, typically a pulsed laser. Captured images are post-processed with algorithms that perform frame-to-frame feature tracking and calculation of flow velocity fields.

PIV is also a useful method for measuring soil motion, with the notable constraint that soil is typically observed through a glass sheet, limiting the resulting analysis to plane strain scenarios. The natural granular texture of soils often generates an intensity pattern that can be readily traced by PIV-algorithms, without the use of marker particles. Also, incandescent light can generally be used for illumination.

Granular PIV has recently been employed in several applications, including the analysis of grains in converging hoppers [10], study of flowing granular layers in rotating tumblers [11], investigation of granular avalanches [12], analysis of soil motion caused by the movement of animals [13], the study of burrowing behavior of razor clams [3], and in the study of wheel-soil interaction [4, 14]. The analysis of soil motion beneath a driven wheel via quantitative analysis of successive temporal images was first introduced by Wong [15]. However, the experimental capabilities of that study did not allow for high-speed image capture, limiting the accuracy and practical utility of the method.

Soil motion analysis can be broken down into four main steps: 1) image acquisition, 2) image

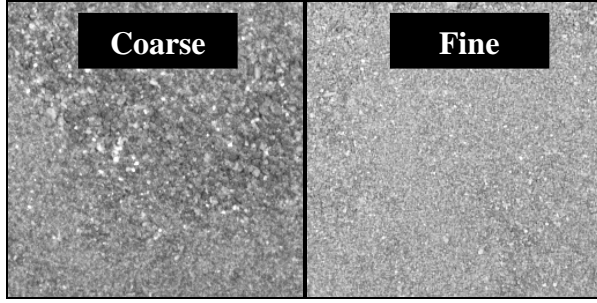


Figure 2: Examples of soil natural textures.

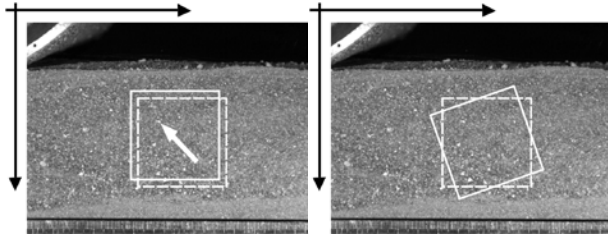


Figure 3: Two examples of image canonical transformations used to evaluate PIV settings. Nine image transformations for coarse and fine soil textures were used to evaluate PIV accuracy.

of 800x600 pixels at a maximum frame rate of 6688 fps. The camera was placed perpendicular to the front glass wall (see Figure 1) at a distance of 52 cm, while the focal length was set to 77 mm (a zoom lens was used) resulting in an image capture region of approximately 15 x 11.25 cm. It should be noted that determination of image capture region size is largely dictated by the particular experimental conditions. Here, the image capture region was chosen in order to conservatively bound the region of soil that would undergo motion when subjected to wheel passage on the soil surface. Two 250W Lowel Pro-Light photography flood lights were placed on either side of the camera at an angle of 45° towards the object plane, and provided approximately homogeneous illumination of the soil. By using two laterally positioned light sources, reflections and shadows can be significantly diminished.

14 *Piv Image Preprocessing*

The performance of PIV cross-correlation algorithms generally improves when images are of high contrast, feature dense, and have low noise. In practice, images are subject to nonuniform illumination, image sensor noise, and lack of natural contrast in the granular material, all of which can degrade PIV algorithm performance. Various image pre-processing methods were investigated to understand their effect on algorithm performance. These include commonly-employed algorithms such as contrast limited adaptive histogram equalization, high pass filtering, and clipping and intensity capping.

To systematically investigate the effect of these preprocessing methods on PIV algorithm performance, test image segments of the Mars regolith simulant with dimensions 256 x 256 pixels were captured, then synthetically deformed in canonical directions. Since the particle distribution in the soil under investigation is locally inhomogeneous, two distinct image

13 *Piv Imager Configuration*

The accuracy of PIV strongly depends on the quality of the captured images. For these experiments the testbed was fitted with a 2.54 cm thick tempered glass wall while the running gear was operated flush against this surface (see Figure 1). Both wheels and tracks have been analyzed with this testbed, however this report describes results from rigid wheel testing.

Image sets for the PIV measurement were captured with a Phantom 7 high-speed camera. The Phantom 7 is able to record grayscale images at the maximum resolution

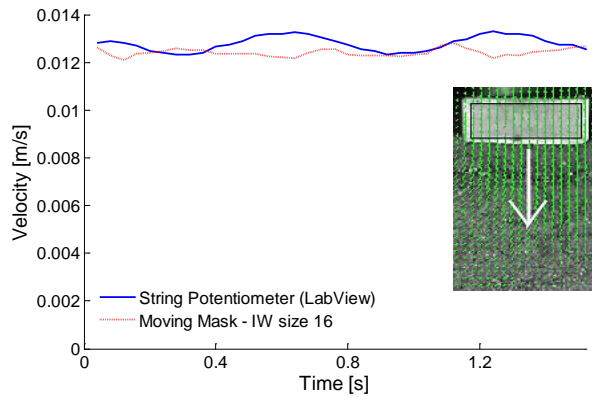
segments were captured in order to adequately represent typical apparent grain distributions in the MMS simulant. This resulted in one image populated by relatively large grains and one populated by relatively small grains (Figure 2). Synthetic deformation of the image was performed as a means of generating a ground truth for cases of linear translation (1-4 pixels in both horizontal, vertical, and diagonal directions), rotation (1-8 degrees in clockwise and counter-clockwise directions), shear (1-4 pixels of relative motion between upper and lower image halves), and simple shear (1-4 pixels of motion of upper edge of image) (Figure 3). Since the pixel shift for each deformation was controlled, this methodology allowed quantitative evaluation of PIV algorithm results. An error metric was computed by computing the average difference, over all points in the PIV velocity field, between the velocity vector calculated through PIV and the true velocity vectors.

15 Piv Image Cross-Correlation

In PIV, images are divided into small interrogation windows (IW) and then analyzed to compute the probable displacement between successive images for each IW using cross-correlation techniques. This results in an equally spaced field of calculated velocity vectors. The probable displacement is determined by using the cross-correlation function:

$$R_{II'}(x, y) = \sum_{i=-K}^K \sum_{j=-L}^L I(i, j) I'(i + x, j + y) \quad (1)$$

where I is the intensity of the first image and I' the intensity of the second image. A detailed description of PIV theory can be found in [17]. Particle density, image resolution, and IW size



are interconnected parameters that must be carefully selected to optimize performance. Based on experimental investigations, Keane and Adrian [18] defined empirical rules for optimal PIV setup. The reader is referred to the above report for more details. For the results presented here, the following settings were employed: 25 fps, final IW size of 16, CLAHE filtering with kernel size of 40 pixels. A more complete description of the PIV settings and analysis is presented in [19].

Figure 4: Comparison of velocity calculated through PIV and measured with a draw wire encoder

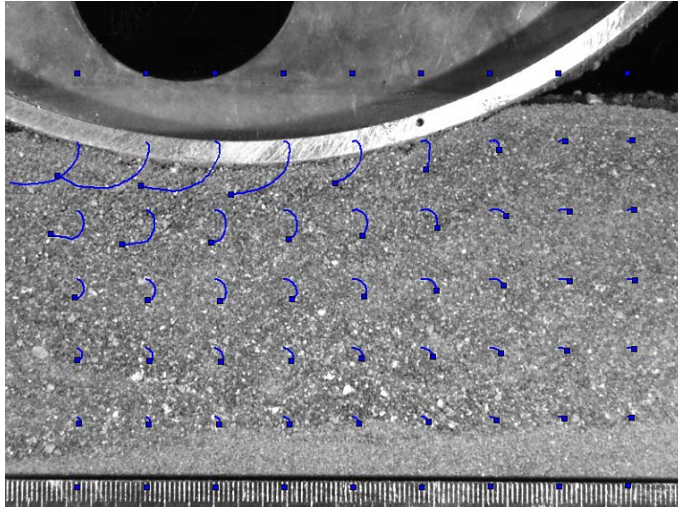


Figure 5: Soil trajectories calculated from velocity field obtained through PIV analysis. Visual inspection showed that PIV yielded tracking of soil regions on the order of 0.5-1 mm after translations of several centimeters.

16 Velocity Field Post-Processing

The raw velocity field produced by PIV calculations can contain spurious vectors (outliers). These outliers can be caused by noise, inappropriate interrogation settings, and accidentally matched patterns. Hence, to improve results, rejection of these outliers and interpolation of missing data points can be performed in a post-processing stage through filtering. Filters for the rejection of outliers can primarily be divided into two separate classes: global and local methods. Global filters commonly employ a simple thresholding method, with the threshold value selected by an operator with empirical or theoretical domain knowledge. If elements of the velocity field exceed the threshold, this element is removed from the results.

Local filters are primarily based on relative differences between surrounding vectors, rather than absolute values. A local filter calculates the mean and standard deviation of the velocity for a selected kernel size around each vector. If the velocity exceeds certain thresholds, the vector is rejected. For the results presented here, a 5x5 kernel with a threshold of 8 times the standard deviation was used for post-processing.

17 Validation And Verification

The synthetically deformed image was determined to be a useful ground truth for determining appropriate PIV operational parameters. However, validation of the PIV algorithm performance was also pursued on two sets of test data that were physically relevant to the running gear-soil interaction case.

The first test consisted of calculating the velocity via PIV of a 2.5 cm thick steel plate performing a soil penetration test. The ground truth velocity of the plate was externally measured by numerically differentiating the output of the draw wire encoder (which nominally provides a position measurement). To obtain a plate velocity measure from PIV, an average of the velocities was computed over a rectangular region of interest aligned with the moving plate.

Figure 4 shows a comparison of the plate velocity as determined from PIV calculations and the velocity measured by the draw wire encoder. The average percent error (for the best settings) between these measurements was below 1%. It should be noted that, for this test case, the PIV algorithm is not performing calculations on the granular soil, but rather the steel plate edge. However, this test remains of interest since the soil in contact with the plate necessarily moves at the same velocity.

The second test consisted of calculating the time evolution of motions of discrete features associated with MMS simulant soil beneath a driven rigid wheel. Trajectories $\mathbf{s}(t)$ are calculated

for a grid of 9 x 6 regions of interest over the soil area. The time evolution of the positions of the center of the regions of interest was computed by integrating the velocities with a fourth order Runge-Kutta method.

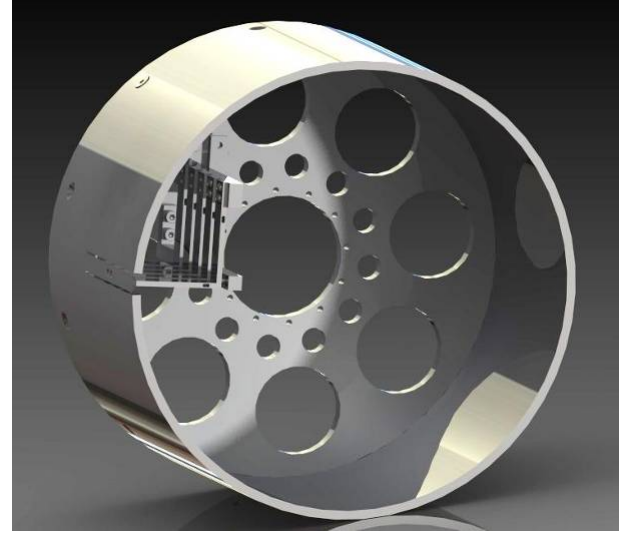
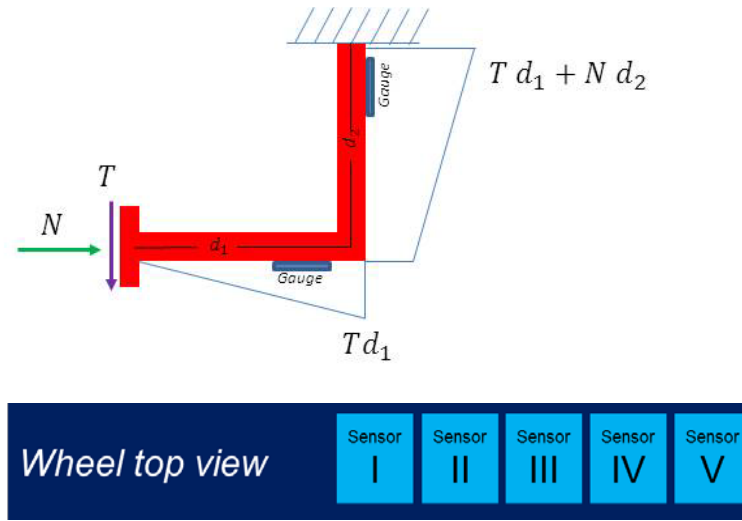


Figure 6: Working scheme of the custom force sensor for interfacial stress measurement (top left). Five sensors are distributed from the wheel median axis to the wheel edge (bottom left). Sensors are rigidly connected to the wheel hub (right)

$$s(t) = \int_n^t v(t)dt \quad (2)$$

The motion of these tracked regions were compared to trajectories of individual soil particles that are large enough to be manually tracked from frame to frame, thereby providing a qualitative performance evaluation. Also, the calculation of feature trajectories is useful for illustrating soil flow when subjected to various loading conditions.

Figure 5 displays the trajectories computed while the wheel was advancing at 17 deg/s with 30% slip. Note that the area above the soil surface was masked during pre-processing, and hence these features remain at their original location. The squares show the final position of the tracked features and the lines represent the motion evolution. Manual inspection showed that the selected PIV yielded tracking of soil regions in the order of 1-2 pixels, corresponding to 0.5-1 mm after translations of several centimeters.

18 Wheel-Terrain Interface Force Sensor Description

Measurement of the normal and shear stress acting on a moving wheel is important for empirical testing and validation of models describing interfacial phenomena. While numerous COTS sensors exist for measuring pressure [20], the authors are unaware of any available sensors that can measure both pressure and shear stress, at a scale and resolution suitable for

investigation of the interaction mechanics of small, lightweight vehicle running gear and deformable soil.

Therefore, a custom sensor array was designed and fabricated (Figure 6). Each sensor is a solid-state L-shaped aluminum flexure instrumented with two full bridge strain gages. The sensor is mounted rigidly to the running gear, and its interface element is exposed to the soil. The interface element is generally subjected to normal (N) and shear (T) loading. These forces cause the flexure elements to deflect in a linear elastic manner. From measured deflection, and given prior calibration data, the applied forces can be uniquely computed. (Axial strain is intrinsically rejected by the full bridge configuration.) Stress can then be inferred assuming uniform pressure distribution over the known sensors' head area.

Sensors are mounted on the surface of a 26 cm diameter rigid aluminum wheel (see Figure 6). Note that a twin wheel, without the array, was used for PIV testing. Five sensors have been fabricated and integrated in a linear array spanning one half of the wheel width (i.e. from one edge to the center of the wheel). Sensors were first calibrated by applying test weights of 100, 200, and 500 grams in the normal and tangential direction. Measurement linearity error, across all the sensors, was found to be below 3%.

The sensor array is extremely sensitive to misalignment and thus an uneven contact patch profile can easily unbalance the output reading. To ensure accurate alignment, sensors alignment was verified after every 5 tests, by driving the wheel over a flat, rigid, aluminum plate covered with a thin layer of polyurethane foam in order to verify that the sensor output was uniform. Due to the difficulty in precisely controlling soil preparation, each test was repeated at least 15 times. In fact, local soil density variation, inhomogeneity (due to non-uniform distribution of larger grains, for example), and surface unevenness all were observed to affect measurement output. The 15 trials highlighted test variability and were analyzed to detect outliers and eventually remove tests where anomalies were detected.

19 Soil Properties

Characterization of the soil under investigation is a necessary step for any terramechanics investigation. Detailed chemical composition, particle size distribution, and shearing properties of the MMS simulant under investigation can be found in [6]. However, pressure-sinkage properties (i.e. Bekker's parameters) for the soil were unknown, and therefore a series of plate penetration tests were performed.

Since the wheel has a width of 0.13 m and a nominal contact patch length of 0.05 m (estimated assuming nominal conditions of $Fz = 100$ N and low slip) three rectangular plates with the following dimensions were selected: 0.13 m x 0.03 m, 0.13 m x 0.05 m, and 0.13 m x 0.07 m.

Each plate was mounted on a linear actuator, which was anchored to the testbed and then pushed perpendicularly into the soil while the vertical load and penetration length (i.e. sinkage) were measured with a load cell and a draw wire encoder, respectively.

For each plate, tests were repeated 15 times. Between each test, soil was manually agitated and then re-leveled. Figure 7 shows an example of the data collected. Test-to-test variation was observed, but was not considered unusual due to the nondeterministic nature of soil testing.

The scope of the tests was to fit experimental data to Bekker's pressure-sinkage equation [21]:

$$p = \left(\frac{k_c}{b} + k_\phi \right) z^n \quad (3)$$

where p is pressure, z is sinkage, b is plate width (3,5,7 cm) and $\{k_e, k_{ph}, n\}$ are the parameters under investigation. Adopting the fitting methodology presented in [2] it was noted that $k_{eqb} = \left(\frac{k_e}{b} + k_{ph}\right)$ is strongly correlated with n as shown in Figure 8. This correlation necessarily results from the tests having similar amounts of deviation from an exponential curve. While this effect is solely an artifact of experimental estimation, it is still undesirable because it inhibits k_{eqb} from being estimated independently.

The problem is mitigated through adoption of Reece's equation [22] for pressure-sinkage:

$$p = k_{eqr} \left(\frac{z}{b}\right)^n \quad (4)$$

Dimensional analysis of Reece's equation shows that k_{eqr} is not function of n (as it was in Bekker equation). Although variability is still substantial, k_{eqr} estimation becomes less dependent of n as can be seen in Figure 9.

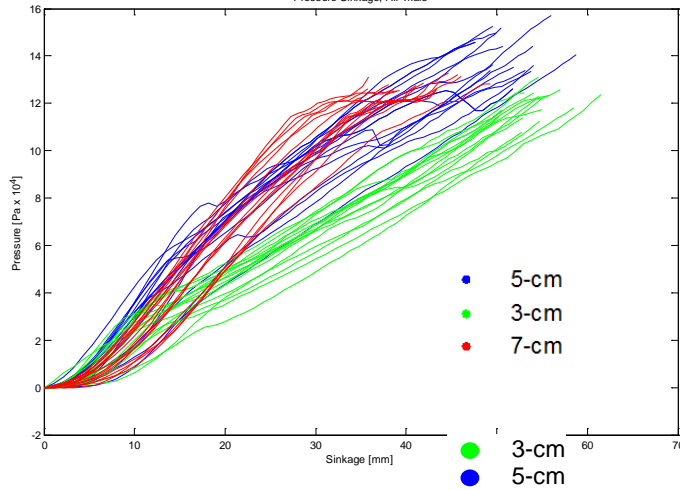


Figure 7: Penetration tests for rectangular plates with the following dimensions 0.13 m x 0.03 m, 0.13 m x 0.05 m, 0.13 m x 0.07 m

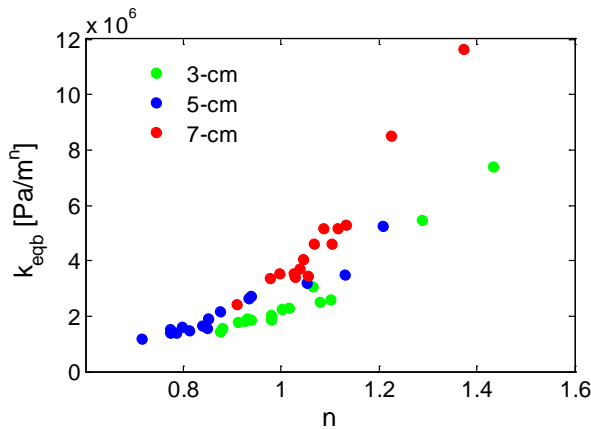


Figure 8: Strong correlation between soil parameters when Bekker equation is used

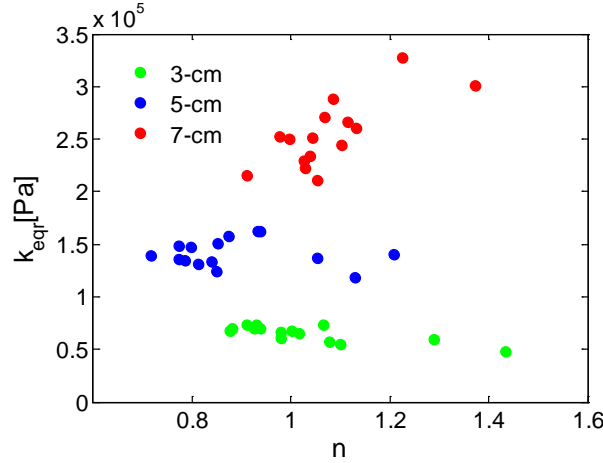


Figure 9: Correlation between soil parameters is mitigated when Reece's equation is used

Penetration tests variability, even under laboratory controlled conditions, suggests that soil parameters should be derived from statistical distributions rather than deterministic values. A stochastic characterization of terrain properties is currently being investigated by the authors while the results presented in this report are still derived with the method established by Wong [2].

Two parameter sets are reported in Table 1. The set labeled '357' has been obtained considering the full dataset presented in Figure 7 while the set labeled "57" has been obtained only with the 5 cm and 7 cm plates, and truncating the data at 50 kPa. This was motivated by the fact that the wheel under investigation was expected to have contact patch length larger than 5 cm and normal stress distribution below 50kPa. The two datasets show how slightly modifying the design of experiments, can drastically change soil parameter calculation.

Table 1: Bekker soil parameters for the MMS soil. Two sets were extracted, 357 includes all the data while for 57 only two plates were used (5 cm and 7 cm) and data was truncated at 50 kPa mark.

Set	n	k_c [kN/m ⁿ⁺¹]	k_ϕ [kN/m ⁿ⁺²]
357	0.99	-55	4584
57	1.4	846	6708

20 Results And Discussion

Experiments with the PIV and stress sensor experimental methodologies were conducted separately. For PIV tests, a smooth wheel, coated with MMS simulant (to ensure sufficient interfacial friction) was run flush against a glass wall. For stress sensor tests, a wheel of exactly the same diameter, and again covered with MMS simulant, was run in the middle of the soil bin. Soil was loosened, mixed, and leveled between each test, in an attempt to achieve uniformly loose, homogenous conditions.

Both type of tests were run at approximately 100N of vertical load and for slip levels ranging from -70% to 70% (for PIV tests, slip was limited to $\pm 30\%$). For PIV tests the wheel velocity was fixed at 17 deg/s while for stress sensor tests angular velocity was reduced to 8.5 deg/s to

improve measurement quality. (The horizontal carriage velocity was modified to achieve the desired slip level.) For both types of tests, it was first ascertained that velocity did not have an influence on wheel performance. The operational conditions described above were chosen because they are close to those of the Mars Exploration Rover, a successful lightweight robotic vehicle.

A substantial amount of data was collected and cannot be comprehensively described in this report. Instead, a small number of initial results are presented.

21 PIV Analysis

Analysis of PIV data was performed to qualitatively analyze soil motion (a quantitative analysis would have required to investigate the complex mapping between stress and displacement, this goes beyond the scope of this preliminary study). Figure 10 presents a snapshot of a 30% slip test, and displays the following information from top-left-clockwise: velocity vectors, u-velocity, v-velocity, and velocity magnitude. Analysis of such images can provide insights into the spatial distribution of soil velocity under running gear, and can vary dramatically for such cases as slip, skid, free-rolling wheels, braked wheels, etc.

Decomposition of this flow field can yield useful insight into soil shearing (which occurs primarily in the horizontal direction, see upper right image) and soil compaction phenomena (which occurs primarily in the vertical direction, see lower right image). Here, a blue region corresponds to no motion while red indicates a maximum velocity. Analysis of these images shows that soil flow remains attached to the wheel rim. Moreover, for low vertical load (such as the one utilized during experiments) it was observed that two separate slip failure lines did not evolve, as predicted by classical theory [23, 24]. This finding is interesting because according to [23], the maximum stress occurs where the soil flow separates. The absence of flow separation, however, does not prevent stress to reach a maximum (see Figure 11).

For slip levels below $\pm 10\%$, the soil was not observed to develop a significant shearing plane. Another phenomenon that was clearly highlighted by PIV analysis is the periodic nature of soil failure. For slip level above 10-15%, soil often exhibits a periodic loading cycle of alternating compaction and shearing, which results in discontinuous failure of the soil mass. This has two direct consequences: oscillations in drawbar pull readings and creation of ripples behind the wheel. Note that while these effects have been noted previously, they have been typically assigned to the effect of grousers. However, these effects are present even for smooth wheels, without grousers.

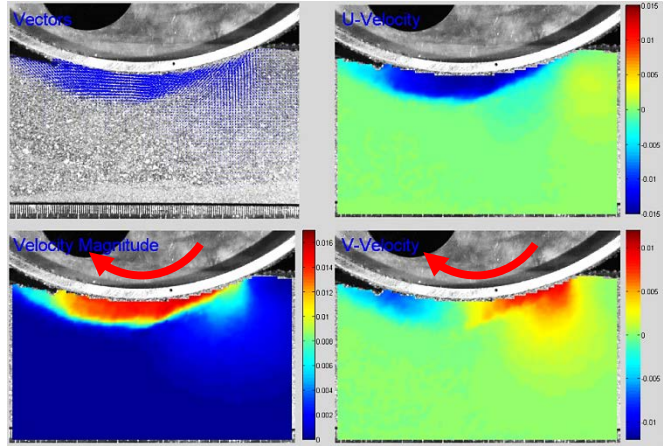


Figure 10: A snapshot of a 30% slip test. Nominal vertical load was 100N and wheel angular velocity of 17 deg/s. From top-left-clockwise: velocity vectors, u-velocity, v-velocity, and velocity magnitude.

PIV data can be useful for investigation of constitutive models for granular materials, and for development of reduced order models based on soil displacement predictions. An important consideration to bear in mind when examining flow fields like the one presented in Figure 10 is that the relationship between stress and displacement is typically complex, and one must avoid the temptation to directly (i.e., proportionally) correlate velocity magnitudes with stress magnitudes.

For this reason, direct stress measurement of shear and normal forces, and inferences of associated stresses, at the wheel-terrain interface yields valuable information about the traction generation process.

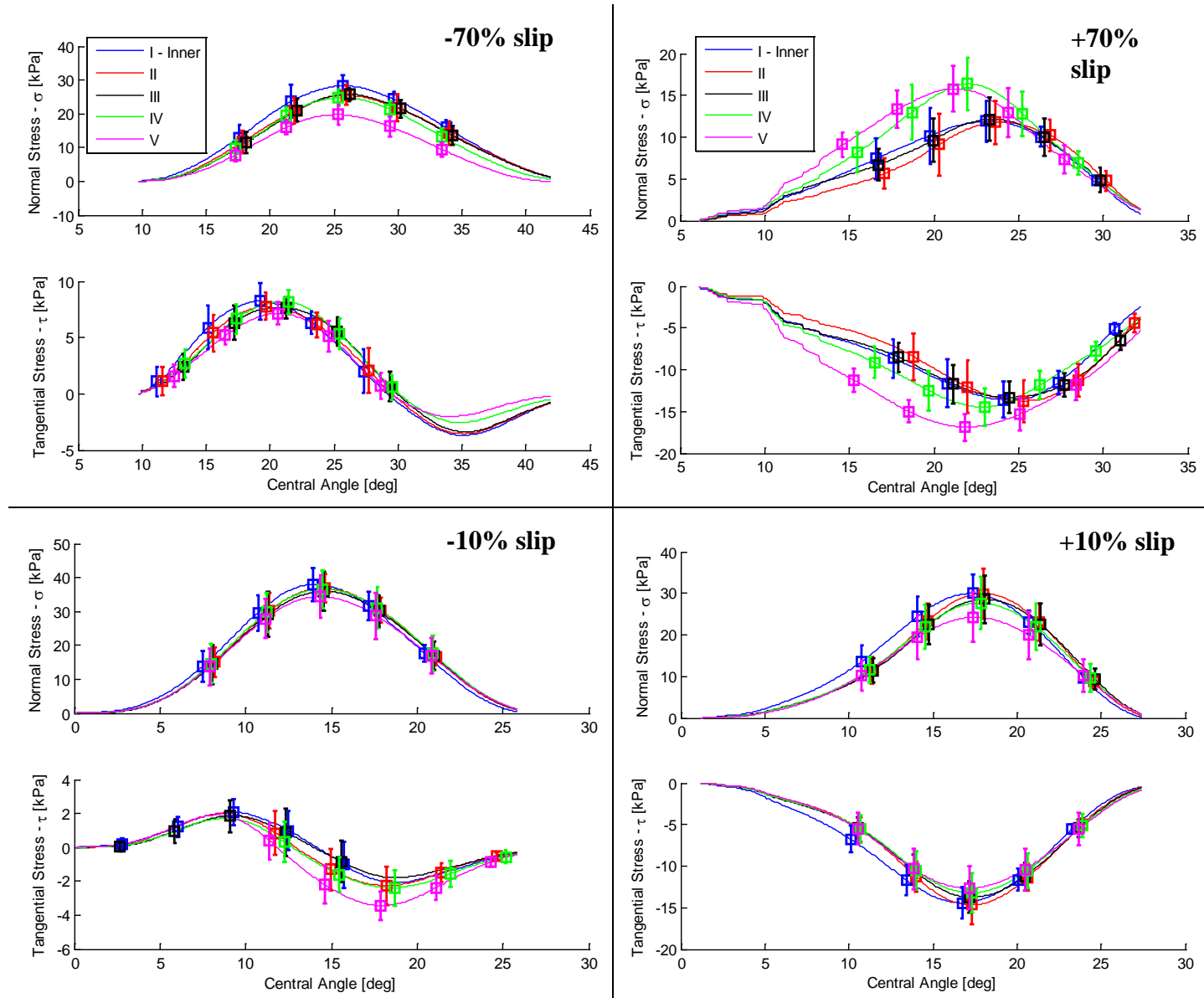


Figure 11: Normal and tangential stress at the wheel-soil interface calculated from force sensors. These were obtained for nominal vertical load of 100 N and wheel angular velocity of 8.5 deg/s. The four panels present data for -70%, +70%, -10%, and +10% slip (clockwise from upper left). Sensors are labeled according to the scheme presented in Figure 6. “I” corresponds to the sensor located at the center and “V” to the sensor located at the edge of the wheel. Central angle defines the angular position along wheel circumference [26,27].

22 Interface Force Sensor Analysis

Classical terramechanics methods rely on the estimation of the stress distribution under the wheel. The ability to directly measure such quantities allows for a one-to-one comparison of model prediction and experimental reality.

Analysis of stress distribution across a (symmetric) half-wheel width shows that boundary effects become more pronounced as slip increases (see Figure 11). In particular, stress at the wheel edge was observed to be relatively high for positive slip and relatively low for negative slip. It is hypothesized that this effect is caused by soil transport phenomena: for positive slip, soil in the center of the wheel is transported behind the wheel at higher rate than the soil at the wheel edges, which causes the wheel edges to bear proportionally more of the total normal wheel load. On the contrary, for negative slip, soil accumulated in front of the wheel creates a thicker layer under the wheel median axis, causing higher stress in the center.

For higher loading conditions, Onafeko and Reece [25] noted that normal stress decreases with

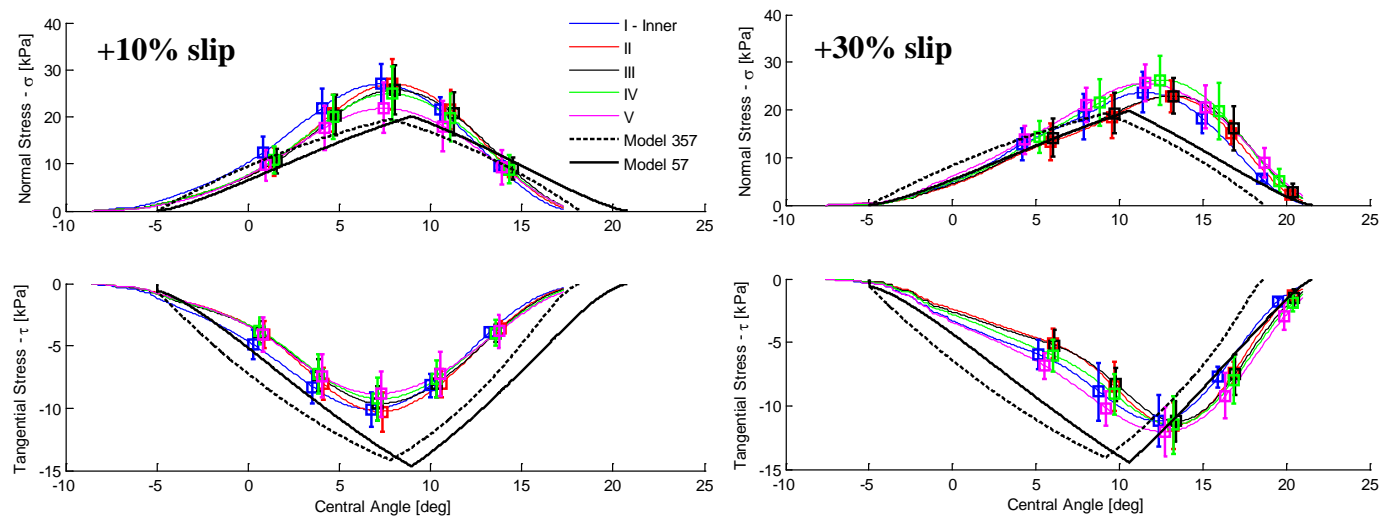


Figure 12: Stress distribution for 10% (left) and 30% (right) slip compared with analytical model from Wong and Reece [26, 27]. Two soil parameter sets, presented in Table 1, were tested. The difference between the two parameter sets, although significant, it is not dramatic. Normal stress is slightly underestimated while tangential stress is significantly estimated. Tangential stress, however, is primarily based on soil shear properties which were obtained in [29].

increasing positive slip since an increasingly larger portion of vertical load is supported by shear stress (which contributes more to vertical load equilibrium because of increased sinkage). This was confirmed experimentally with the stress sensors.

Another interesting aspect of wheel stress distributions is the inversion of shear stress for negative slip conditions. This phenomenon was noted also by [25] and it is consistent with wheel-soil interaction kinematics: for negative slip, the wheel travels forward but simultaneously skids over the soil, generating a shear sign transition. Interestingly, PIV imagery does not show any soil separation or flow inversion where the shear stress changes sign.

In Figure 12, a direct comparison between the measured stress and stress predicted by the model originally proposed by Wong [26, 27] and Janosi and Hanamoto [28] is presented, using the experimentally determined soil parameters (two parameter sets, presented in Table 1, are compared). The normal stress distribution is underestimated and the error seems largely related

with the location of maximum stress. Tuning of semi-empirical model parameters could allow better agreement.

The predicted shear stress, however, was found to be overestimated. Note that the shear modulus adopted to produce results in Figure 12 was calculated according to [29]. For larger (but arguably inaccurate) values of shear modulus, it may be possible to obtain better agreement between prediction and experimental data; however this raises a fundamental question about the validity of the assumptions behind the model. In fact, the model assumes that the soil is sheared for a distance corresponding to the amount of relative motion between the wheel and the soil. This assumption, as shown by PIV analysis, is likely erroneous, since the soil at the wheel-terrain interface stays attached to wheel rim, while failure physically occurs (in regular, periodic failure patterns) some distance away from the interface. Although $\{n, k_c, k_\phi\}_{357}$ and $\{n, k_c, k_\phi\}_{57}$ are significantly different (see Table 1), model predictions using these two sets are relatively close. This warrants further efforts in characterizing terrain variability and its influence on stress measurements variability.

23 Conclusions—Experimental Analysis of Wheel-Terrain Interaction

Novel experimental methods aimed at understanding the fundamental phenomena governing the motion of lightweight vehicles on dry, granular soils were presented.

Aside from standard wheel experiments (i.e., measurements of drawbar force, applied torque, and sinkage during controlled slip runs) two additional experimental methodologies were introduced. The first relies on high-speed imaging of the wheel-soil interface and the use of particle image velocimetry (PIV) to measure micro-scale terrain kinematics. The second experimental methodology consists of a custom force sensor array located at the wheel-terrain interface. The sensors allowed explicit measurement of normal and shear forces (and, therefore, estimation of normal and shear stresses) at numerous discrete points along the wheel-soil interface.

Analysis of PIV data has shown that soil failure, at certain slip levels, is qualitatively different under cases of low vertical load (which is typical for lightweight robots) compared to cases of high vertical load (typical for large ground vehicles). Also, soil flow patterns have been observed to exhibit periodic failure phenomena, giving rise to interesting features such as surface ripple formation. Soil flow was observed to be always attached to the wheel rim and only one shear failure surface was observed. Soil usually exhibits compression in front of the wheel and then shears beneath it.

Stress measurements showed that, although only one shear failure surface is present, tangential stress goes through sign inversion for negative slip. Stress distribution, along the wheel width, is approximately uniform for low slip while edge effects become increasingly significant for higher slip levels. Although some observations regarding soil shear failure were not confirmed by PIV, classical methods (partially based on those observations) were able to capture main trends for a range of slip conditions. These results provide deeper understanding of the mechanics of traction generation and are expected to open new frontiers for more accurate, and predictive, lightweight vehicle mobility models.

Further investigation of small robot-terrain interaction mechanics will focus on extending these experiments to a wider range of vertical loads. This will provide a basis for validation of constitutive laws and the improvement of reduced-order models. Future work will also focus on stochastic characterization of terrain response and how underlying soil variability affects interfacial stresses modeling. In fact, even under laboratory controlled conditions, penetration

plate tests have highlighted significant soil variability, warranting for statistical interpretation of experimental data.

24 REFERENCES

- [1] Meirion-Griffith, G., and Spenko, M., “A modified pressure–sinkage model for small, rigid wheels on deformable terrains,” *Journal of Terramechanics*, Volume 48, Issue 2, pp. 149-155, April, 2011.
- [2] Wong, J.Y., “Data processing methodology in the characterization of the mechanical properties of terrain,” *Journal of Terramechanics*, 17(1):13 – 41, 1980.
- [3] Winter, A., “Biologically Inspired Mechanisms for Burrowing in Undersea Substrates,” Ph.D. Thesis, Massachusetts Institute of Technology, 2010.
- [4] Moreland, S., Skonieczny, K., Wettergreen, D., “Soil Motion Analysis System for Examining Wheel-Soil Shearing,” 17th International Conference of the International Society for Terrain-Vehicle Systems, Virginia, USA, 2011.
- [5] Iagnemma, K., A Laboratory Single Wheel Testbed for Studying Planetary Rover Wheel-Terrain Interaction, Technical Report 01-05-05, Field and Space Robotics Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 2005.
- [6] Beegle, L. W., Peters, G. H., Mungas, G. S., Bearman G. H., Smith, J. A., and Anderson, R. C., “Mojave Martian simulant: a new martian soil simulant,” *Lunar and planetary science*, XXXVIII, 2007.
- [7] Adrian, R. J., “Statistical Properties of Particle Image Velocimetry Measurements in Turbulent Flow, *Laser Anemometry in Fluid Mechanics*,” Lisbon: Instituto Superior Tecnico, 1988, pp. 115–29.
- [8] Keane, R. D., and Adrian, R.J., “Theory of Cross-Correlation Analysis of PIV Images,” *Applied Scientific Research*, Vol. 49, Netherlands, 1992, pp.191-215.
- [9] Adrian, R. J., “Twenty Years of Particle Image Velocimetry,” *Experiments in Fluids*, Vol. 39, 2005, pp. 159–169.
- [10] Sielamowicz, I., Błoński, S., and Kowalewski, T. A., “Digital Particle Image Velocimetry (DPIV) Technique in Measurements of Granular Material Flows,” *Chemical Engineering Science*, Vol. 61, 2006, pp. 5307 – 5317.
- [11] Jain, N., Ottino, J. M., and Lueptow R. M., “An Experimental Study of the Flowing Granular Layer in a Rotating Tumbler,” *Physics of Fluids*, Vol. 14, 2002, pp. 572-582.
- [12] Pudasaini, S. P., Hsiau, S. S., Wang, Y., and Hutter, K., “Velocity Measurements in Dry Granular Avalanches Using Particle Image Velocimetry - Technique and Comparison with Theoretical Predictions,” *Physics of Fluids*, Vol. 17, 2005.
- [13] Barnett; C. M., Bengough, A. G., and McKenzie, B. M., “Quantitative Image Analysis of Earthworm-Mediated Soil Displacement,” *Biology and Fertility of Soils*, 2009, Vol. 45, pp. 821-828.
- [14] Saengprachatanarug, K., Ueno, M., Taira, E., and Okayasu, T., “Modeling of Soil Displacement and Soil Strain Distribution Calculation under the Traveling Wheel,” 17th Int. ISTVS Conference, Blacksburg, Virginia, 2011.

- [15] Wong, J. Y., "Behaviour of Soil beneath Rigid Wheels, *Journal of agricultural Engineering*," Vol. 12(4), 1967, pp. 257-269.
- [16] Thielicke, W., and Stamhuis, E. J., "PIVlab – Time-Resolved Digital Particle Image Velocimetry Tool for Matlab," 2010.
- [17] Adrian, R. J., and Westerweel, J., Particle Image Velocimetry, Cambridge University Press, New York, 2011.
- [18] Keane, R. D., and Adrian, R. J., "Theory of Cross-Correlation Analysis of PIV Images," *Applied Scientific Research*, Vol. 49, Netherlands, 1992, pp.191-215.
- [19] Wulfmeier, M., "Development of a Particle Image Velocimetry Method for Analysis of Mars Rover Wheel-Terrain Interaction Phenomena", B.S. Thesis, Gottfried Wilhelm Leibniz Universitaet Hannover, 2012.
- [20] Nagatani, K., Ikeda, A., Sato, K., and Yoshida, K. "Accurate estimation of drawbar pull of wheeled mobile robots traversing sandy terrain using built-in force sensor array wheel," *Proceedings of the International Conference on Robots and System*, pp. 2373-2378, St. Louis, MO, October, 2009.
- [21] Bekker, M. G., Theory of Land Locomotion, University of Michigan Press, Ann Arbor, MI, 1956.
- [22] Reece, A. R., "Principles of soil-vehicle mechanics," *Proc. Instn. Mech. Engrs.*, Vol. 80, Pt 2A No 2, 1965.
- [23] Wong, J. Y., Theory of ground vehicles - 3rd ed., 528 p., New York, Wiley, 2001.
- [24] Karafiath, L. L. and Nowatzki, E. A., Soil Mechanics for Off-Road Vehicle Engineering, Trans Tech Publications, Series on rock and soil mechanics Vol.2, No.5, Clausthal, West Germany, 1978.
- [25] Onafeko, O., and Reece, A. R., "Soil Stresses and Deformations beneath Rigid Wheels," *Journal of Terramechanics*, Vol. 4(1), 1967.
- [26] Wong, J. Y., and Reece, A. R., "Prediction of rigid wheel performance based on the analysis of soil-wheel stresses part I," *Journal of Terramechanics*, Vol. 4, No. 1, pp. 81-98, 1967.
- [27] Wong, J. Y., and Reece, A. R., "Prediction of rigid wheel performance based on the analysis of soil-wheel stresses part II," *Journal of Terramechanics*, Vol. 4, No. 2, pp. 7-25, 1967.
- [28] Janosi, Z., and Hanamoto, B., "Analytical determination of drawbar pull as a function of slip for tracked vehicles in deformable soils," 1st International Conference on Terrain-Vehicle Systems, Turin, Italy, 1967.
- [29] Senatore, C., and Iagnemma, K., "Direct Shear Behavior of Dry, Granular Soils for Low Normal Stress with Application to Lightweight Robotic Vehicle Modelling," *Proceedings of the International Symposium of the International Society of Terrain-Vehicle Systems*, 2011.